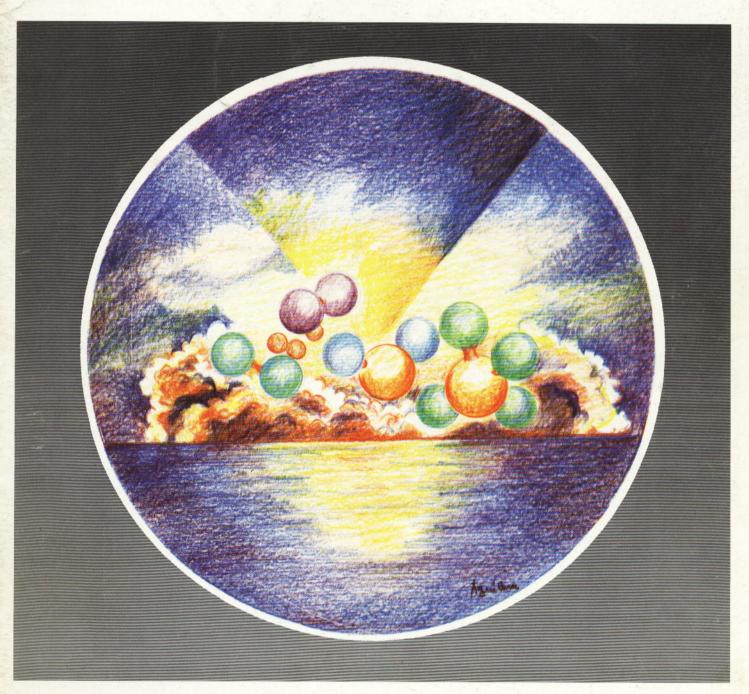
ANO 1
Nº 9
JUNHO 1989
NCz$ 2,70

# CPU

## ELEMENTOS QUÍMICOS
O MSX trabalhando com elementos químicos

## CURSOR PISCANTE NO MSX
Uma nova abordagem

## TRAFALGAR PLUS
Jogo do tipo Wargame

# MSX PAGE MAKER

A NEMESIS INFORMATICA esta' lancando um software revolucionario. O MSX tera' duas epocas distintas: Antes e depois do:

**MSX PAGE MAKER 1.2**

Mais que um processador, ele reune textos e graficos para o desenvolvimento de paginas super especiais para as mais diversas utilidades. De asas a sua imaginacao:

Os modulos principais do programa sao: O Redator, responsavel por toda a parte escrita; o Desenhista, com recursos graficos ilimitados...

Temos ainda o Estilista, que prepara a pagina a ser editada com diferentes bordas, titulos, cabecalhos, etc. Enfim, o estilo do trabalho.

O MSX PAGE MAKER 1.2 e' ainda compativel com o GRAPHOS III e seus:

## SHAPES

Figuras decorativas que podem ser colocadas facilmente em seus desenhos;

## LETRAS

ALFABETOS, ALFABETOS, ALFABETOS, e diversos tipos de letras disponiveis, inclusive COMPACTADA e INVERTIDA, ou seja, centenas de combinacoes !

## TELAS

Compatibilidade a nivel de telas com todos os editores graficos existentes no mercado para MSX. Telas no padrao ".SCR" e ".GRP" sao carregadas sem problemas ("user-friendly").

**MSX PAGE MAKER 1.2**
**SO EM DISCO — 3 OTN.**

**ÁGUIA INFORMÁTICA LTDA.**
AV. N. S. DE COPACABANA, 605/804
COPACABANA
22040 – RIO DE JANEIRO – RJ
TELEFONE: 021-235.3541

DIRETOR RESPONSÁVEL
**GONÇALO R. F. MURTEIRA**

DIRETOR COMERCIAL
**JOSÉ GERALDO J. NETO**

DIRETOR ADMINISTRATIVO
**JOSÉ IDEMAR A. NASCIMENTO**

ASSESSORIA TÉCNICA
**DIVINO C. R. LEITÃO**

JORNALISTA RESPONSÁVEL
**DOLAR TANUS**
**REGISTRO 430-RS**

COLABORADORES
**PEDRO HENRIQUE GAMA**
**PAULO MARQUES FIGUEIRA**
**SÉRGIO GUY PINHEIRO ELIAS**
**PAULO ROBERTO PINHEIRO ELIAS**
**BRUNO MARRUT**
**JÚLIO VELLOSO**
**SÉRGIO DURIC CALHEIROS**
**DIVINO LEITÃO**
**GUILHERME A. L. DA SILVA**
**ANDRÉ L. A. SANTOS**
**MARCOS R. TAVARES**
**EDUARDO R. TAVARES**

REVISÃO DE TEXTO
**LAURA MARIA PINTO**

CAPA
**JOSÉ AGUILERA**

PROJETO GRÁFICO
**LÍBERO NUNES NETTO**

ARTE FINAL
**THEÓFILO RODRIGUES MATTOS**

ASSINATURAS
**MARILZA LOURENÇO**

IMPRESSÃO
**PONTUAL PAP. E IND. GRÁFICA LTDA.**

DISTRIBUIÇÃO
**FERNANDO CHINAGLIA DISTRIBUIDORA**

ASSESSORIA DE MARKETING
**OFFICINA DE CRIAÇÃO E MARKETING LTDA.**



# PAUTA

É com enorme satisfação que lhes apresentamos este novo número de CPU, onde, mais uma vez, foram feitas mudanças.

Novas seções foram introduzidas e outras receberam um novo enfoque.

Isto porque, sem pressa, estamos crescendo e, cada vez mais, podemos apresentar um trabalho que possa atender a todos os nossos leitores.

Este mês, inauguramos o CBBS da revista, aberto a todos os nossos leitores ou não, que irão contar com inúmeros serviços e com um banco dos progreamas que são publicados em CPU. O CBBS funcionará 24 horas por dia. Maiores detalhes para acesso poderão ser encontrados neste número.

No mês de abril, dando provas de que o MSX 1 ainda tem muito tempo de vida no Brasil, a Gradiente apresentou ao público, na UD 89, seus novos produtos de informática que, se realmente lançados, darão um novo grande impulso à linha. Pelo que foi mostrado, vale a pena aguardar até outubro, mês para o qual estão previstos os lançamentos. Na área de software, grandes lançamentos também são esperados. Entre eles o Aquarela, um poderoso editor gráfico, que está sendo elaborado há um ano e que será lançado em junho.

É também na área de software que as coisas andam fervilhando. A época dos avisos e dos anúncios pedindo informações sobre piratas acabou e, agora, as softhouses partem para ação. Na linha do PC, a Black & Decker foi acionada judicialmente, por utilizar em seus escritórios cópias não autorizadas de programas. Na linha MSX, as softhouses já se uniram e, em breve, saberemos quem é quem neste mercado que há muito vem se profissionalizando e que consegue lançar poucos mas significativos softwares de qualidade nacionais, mesmo vendendo jogos.

Tudo isto nos leva a crer que o MSX ainda tem muita estrada pela frente, e que as obras nesta estrada foram retomadas para tornar a caminhada do usuário mais suave e vantajosa.

Portanto, caro leitor, ligue o seu micro e aprecie este número de CPU, pois temos muito ainda para descobrir e aprender.

## NOVA LINHA GRADIENTE

Na linha de micros, a Gradiente lançou na UD 89 seus novos micros, do padrão MSX.

O Expert Plus, que irá atender os iniciantes de informática, a Educação e o Lazer, será comercializado com softwares utilitários residentes, para desenho e música. O Expert Plus possui 80 Kbytes de RAM e 48 Kbytes de ROM, sendo que os 16 Kbytes que foram incorporados correspondem aos programas residentes.

O Expert DD Plus, desenvolvido para atender pequenas e médias empresas e profissionais liberais, já vem com um dirve de 3 1/2" embutido no próprio gabinete, com capacidade de armazenamento de 720 Kb formatados.

Os novos micros da linha MSX da Gradiente possuem um novo chip, que congrega o Z80, VDP e PSG e apresentam paginação de memória diferente do Expert 1.1, comercializado atualmente pela Gradiente, sendo que a cor predominante passou a ser a preta.

Na linha de periféricos, além do monitor monocromático de 80 colunas, do gravador Datacorder DR-1 e do Joystick, foram apresentados o Multi-Modem e um cartão de 80 colunas.

O Multimodem permite a troca de mensagens e arquivos entre dois micros MSX ou a troca de arquivos entre um micro MSX e um micro padrão IBM-PC, via linha telefônica comum. Possui discagem via teclado, com monitorização de linha, programas para acesso a videotexto e comunicação micro-a-micro (300/300 baud full duplex) residentes, operando nos principais protocolos de comunicação.

O cartão de 80 colunas é apresentado em duas versões. O CT-80E permite trabalhar com telas de 1 a 80 colunas x 24 linhas e já vem com um poderoso redator de textos residente, que mostra na tela do micro como o texto será impresso, podendo utilizar vários tipos de caracteres.

O CT-80NET vem a ser um cartão de 80 colunas com emulador de terminal IBM-PC residente. Possibilita a programação da RS-232C via Basic Exxtendido e trabalha com telas de texto de 1 a 80 colunas x 24 linhas.

A nova linha de informática da Gradiente será comercializada a partir de outubro próximo.

## RIOSOFT

A RIOSOFT está completando seu 3º aniversário e, devido à dedicação recebida por parte dos clientes e fornecedores, tem o prazer de oferecer, GRATUITAMENTE, um curso de operação de Drive, com o objetivo de instruir a todos os usuários da linha MSX que queiram adquirir maior performance em suas atividades operacionais.

Maiores informações poderão ser obtidas através do tel. 264-3726 ou à Rua Conde de Bonfim, 346 – Loja 107, na Praça Saens Peña – Tijuca.

## ELATEX MULTI SYSTEM: TECNOLOGIA DE ÚLTIMA GERAÇÃO

Desenvolvido com o objetivo de agilizar ao máximo a elaboração de desenhos têxteis em jacquard, chega ao Brasil, através da Santista Têxtil, o Elatex Multi System, que vem a ser um equipamento computadorizado de tecnologia alemã de novíssima geração.

O sistema é composto de um microcomputador com grande ampliação de memória para comandos centrais, monitor colorido com elevada qualidade de resolução, possibilitando a elaboração econômica e mais racionalizada de cartões perfurados em jacquard.

Em função da grande capacidade de suas memórias externas, um grande número de padronagens pode ser dinamicamente controlado. Assim, padronagens extensas podem ser compostas em uma operação seqüencial, fazendo com que os desenhos apareçam desta forma no monitor. Depois de compor, controlar e corrigir o desenho totalmente, que fica armazenado em um hard disk, produz-se, automaticamente, um gráfico pontilhado em papel, com as orientações adequadas para o tecimento.

A entrada de dados (leitura) e a sua manipulação é feita em dois sistemas independentes, o que garante o melhor aproveitamento e rentabilidade do equipamento no qual foram investidos US$ 340 mil pela Santista. ■

CPU NEWS

## MSX NEWSOFT X AIDS

Dando prosseguimento à comercialização de "softs profissionais", a Newsoft Informática Ltda. lança, com exclusividade, um programa mundialmente inédito sobre a AIDS.

Trata-se de um programa voltado para a comunidade médica, cuja elaboração foi realizada pelo Dr. Pierre G. J. Ciriades, que é médico especializado em Medicina Interna e Patologia Clínica, Chefe do Laboratório Central da AMICO e Diretor Clínico do Hospital AMICO em Indianópolis, tendo sido baseado no trabalho científico de autoria do próprio Dr. Pierre, intitulado "Estimativa da Cronologia e da Depleção Linfocitária na Síndrome da Imunodeficiência Adquirida" – um prognóstico Matemático que recebeu o Prêmio ABRAMGE/88.

O programa interessa não só à prática da medicina diária e à patologia clínica, bem como às instituições seguradoras, previdenciárias, medicina do trabalho e medicina legal, pois, com o mesmo, pode-se calcular a data estimada do contágio inicial pelo HIV, data provável de confirmação do diagnóstico, data limite de sobrevivência à doença, tempo máximo de incubação do HIV, tempo máximo de duração da doença propriamente dita, tempo máximo entre o contágio inicial e o óbito e a estimativa do número absoluto de linfócitos do tipo T4-helper em qualquer data desejada, usando como dados necessários duas contagens sucessivas do número absoluto dos linfócitos T4-helper e suas respectivas datas.

Maiores informações poderão ser obtidas na Newsoft.

## ELITE INFORMÁTICA

Em Niterói há uma nova software, que além de comercializar programas, tem revistas CPU, programas originais da Paulisoft, promovendo, para seus clientes, sorteios de brindes diversos. O endereço da Elite é Caixa Postal 101.016 – Niterói – RJ – CEP 24000.

## PAULISOFT NEMESIS E XSW

A Paulisoft, Nemesis e XSW estavam presentes na UD 89, no stand da Gradiente,onde apresentaram seus programas ao público.

Em matéria de software, o grande lançamento da feira foi, sem dúvida alguma, o editor gráfico Aquarela, que foi considerado um dos melhores editores gráficos existentes para MSX, possuindo excelentes recursos. O Aquarela será comercializado pela Paulisoft a partir do próximo mês.

A Nemesis e a XSW apresentaram os programas que já são conhecidos dos leitores de CPU, como o MSX Page Maker, MSX Charter, VOX, Fluxo de Caixa, etc.

## CURSO DE BASIC EM VÍDEO

Com o sucesso obtido com o lançamento da fita de vídeo "Dominando o MSX", a MPO Vídeo está colocando no mercado de informática outro produto inédito.

Trata-se da fita "Curso de Basic MSX", que ensina os interessados no tema a se tornarem programadores de linguagem Basic. A fita é acompanhada de um livro, que traz exercícios para serem resolvidos pelos programadores, além da explicação teórica da linguagem.

Com 80 minutos de duração, o "Curso Basic MSX" abrange programas de operações financeiras, desenhos gráficos, construção de desenhos animados, comandos musicais por computador, oferecendo, ainda, dicas para o dia-a-dia de um programador, tais como: travar e destravar programas, fazer cópias de programas travados, etc.

Como na primeira fita lançada pela MPO Vídeo, o professor é o Prof. Pierluigi Piazzi.

A fita pode ser encontrada nos grandes magazines e em softhouses, como a Paulisoft, Ectron, Nemesis, entre outras. Maiores informações podem ser obtidas pelo telefone 011-285-03875, ou através de correspondência para Av. Paulista 2001 – 19º andar – Conjunto 1923 – Cerqueira César – 01311 – São Paulo – SP.

BASIC

# ELEMENTOS QUÍMICOS

*NORBERTO TSOULESSKI*

A Química está fundamentada nos elementos químicos e suas características distintas, tais como a sua distribuição eletrônica e a sua massa. O programa dado a seguir trabalha com elementos químicos e serve aos estudantes de 1º e 2º graus na área de física e química e trata, exatamente, o seguinte:

F1 – desenho na SCREEN 2 da distribuição dos elétrons em suas camadas.

F2 – diagrama de Linus Pauling.

F3 – ficha do elemento químico, com informações sobre:
a) símbolo do elemento químico,
b) nome do elemento químico,
c) número atômico do elemento,
d) massa do elemento químico,
e) número de nêutrons do núcleo,
f) número de camadas do átomo,
g) valência,
h) família ou grupo a que pertence o elemento.

F4 – volta para a tela principal.

F5 – distribuição dos elétrons em subníveis energéticos através do diagrama de Pauling.

F6 – sai do programa.

Uma característica importante do programa apresentado é o uso de interrupções acionadas pelas teclas de função do MSX (function keys) que, aliás, são uma das muitas particularidades que diferem os micros do padrão MSX das outras máquinas. O modo como me utilizei dessa vantagem é pouco comum e merece explicação

Geralmente, as interrupções geradas pelas function keys são usadas da seguinte maneira: primeiro, são direcionadas através de instrução "ON KEY GOSUB (lista do número da linha)", que indica qual é a subrotina que deve ser seguida em caso de ser pressionada uma das teclas de função. Depois, é necessário ativar a interrupção através da instrução "KEY (X)", onde X é o número da tecla de função que, a partir do momento que o micro receber essa instrução, vai gerar uma interrupção ao ser acionada. Assim, toda vez que pressionarmos uma das teclas de função e esta estiver habilitada a gerar interrupção, o programa será desviado para a subrotina encarregada da tecla de função selecionada. Essa subrotina será executada na íntegra e abortada no instante em que o micro encontrar uma instrução "RETURN".

Até aqui, todos os usuários que realmente conhecem o Basic do MSX não terão tido nenhuma exclamação. A partir de agora é que o negócio fica interessante. Suponha que você deseje sair de uma subrotina chamada por uma function key apertando outra tecla de função antes que a subrotina termine, ou seja, antes que o programa encontre uma instrução "RETURN". Como fazer? É simples, embora não tão óbvio. Basta que, dentro de todas as subrotinas que são chamadas através de teclas de função, nós tornemos a ativar as interrupções novamente. Assim, existirá sempre uma tela principal ou rotina principal que se encarregará de descrever o que faz cada tecla de função, direcionar e habilitar as interrupções para as subrotinas que poderão terminar com uma linha do tipo:

```
1000 GOTO 1000
```

Esta linha faz o programa parar esperando uma tecla de função que, depois de pressionada, desvia o programa para outra rotina do programa. O fato de termos habilitados duas vezes as function keys, fazem com que elas tenham prioridade total, podendo interromper o programa em qualquer ponto. É isso que eu queria que todos entendessem. De resto, é só ligar o programa e comprovar na prática o que eu acabei de explicar.

O PROGRAMA

A digitação é bastante simples, bastando inserí-la tal como está na listagem.

Ao rodá-lo, o usuário verá, inicialmente, um menu com opções e serem selecionadas através das function keys. Estas opções já foram descritas no início deste artigo. Nessa tela é pedido também o número do elemento químico. A seguir, o MSX apresenta a ficha desse elemento. Pressionando F5, teremos a divisão em subníveis energéticos. E, por último, pressionando F1, o desenho da distribuição dos elétrons na eletrosfera. A todo instante que quiser (pelos motivos já mencionados), você poderá observar a ficha ou a distribuição dos elétrons teclando uma function key.

O programa de Química para o MSX foi escrito totalmente em Basic e tem a seguinte estrutura:

– Linha 50 a 170 – tela principal
– Linha 190 a 230 – busca através do número atômico
– Linha 240 a 280 – busca através do nome do elemento
– Linha 290 a 430 – algoritmo para a distribuição dos elétrons em subníveis energéticos
– Linha 440 a 580 – determinação da família a que pertence o elemento químico
– Linha 590 a 740 – tela que mostra, como num quadro negro, a distribuição em subníveis energéticos
– Linha 790 a 1030 – rotina que desenha um átomo com os elétrons convenientemente distribuídos pelas suas camadas.
– Linha 1030 a 1460 – diagrama de Linus Pauling
– Linha 1470 a 1660 – ficha do elemento químico
– Linha 1670 a 2190 – linhas data com os dados de cada elemento.

Observe que incluí nessas linhas os dados absolutamente necessários para cada elemento químico. Os outros dados, como a distribuição eletrônica, o número de nêutrons, o número de camadas do átomo, a valência e a família a que pertence o elemento químico são calculados pelo programa não fazendo parte, portanto, das linhas DATA.

– Linha 2210 a 2300 – subrotina que cria sprites aumentados com símbolos dos elementos químicos. Mais outra interessante rotina que merece ser analisada pelos leitores. ■

```
10 '************************************
20 '
30 ' CREATED BY NORBERTO TSOULEFSKI
40 '
50 '************************************
60 CLEAR:DIM F$(20),N(20)
70 ON KEY GOSUB 970,1240,1710,60,740
80 KEY(1)ON:KEY(2) ON:KEY(3) ON:KEY(4)ON
:KEY(5) ON:KEY(6)ON
90 DEFINT A,B,C,D
100 SCREEN0:WIDTH39:COLOR 1,3
110 KEY OFF:RESTORE1910
120 PRINTSTRING$(38,203)
130 PRINT:PRINT"ESTE PROGRAMA TRABALHA C
OM' ELEMENTOS  QUIMICOS DA SEGUINTE FORM
A:"
140 PRINT:PRINT"F1=DISTRIBUICAO EM NIVEI
S ENERGETICOS"
150 PRINT:PRINT"F2=DIAGRAMA DE LINUS PAU
LING"
160 PRINT:PRINT"F3=FICHA COM CARACTERIST
ICAS"
170 PRINT:PRINT"F4=RETORNA A ESTA TELA"
180 PRINT:PRINT"F5=DISTRIBUICAO EM SUBNI
VEIS ENERGETI- COS"
190 PRINT:PRINT"F6=FIM"
200 PRINT:PRINTSTRING$(38,204)
210 LOCATE0,20
220 INPUT"QUAL O  NOME DO ELEMENTO OU  S
EU No A- TOMICO";N$
230 IF LEN (N$)>3 THEN 320
240 '
250 '           PROCURA NA TABELA
260 '
270 FOR I=1 TO VAL(N$)
280 READ E$,S$,M$
290 NEXT
300 Z=I-1
310 GOTO 370
320 FOR I=1 TO 103
330 READ E$,S$,M$
340 IF E$=N$ THEN Z=I:GOTO 360
350 NEXT
360 A$=N$
370 S=Z:RESTORE 2430
380 '
390 '           CALCULA NUM. ELETRONS
400 '
410 FOR E=1 TO 19
420 READ F$(E):Y$=MID$(F$(E),2)
430 IF Y$="s" THEN W=2
440 IF Y$="p" THEN W=6
450 IF Y$="d" THEN W=10
460 IF Y$="f" THEN W=14
470 S=S-W:IF S<0 THEN N(E)=S+W
480 IF S=0 THEN N(E)=W
490 IFS>0 THEN N(E)=W:NEXT
500 FOR X=1 TO E
```

```
510 V=VAL(F$(X))
520 VI(V)=VI(V)+N(X)
530 NA=NA-(V>NA)
540 NEXT:K=NA
550 IF MID$(F$(E),2,1)="s" AND N(E)=1 TH
EN G$="Alcalinos"
560 IF MID$(F$(E),2,1)="s" AND N(E)=2 TH
EN G$="Alcalino-terrosos"
570 IF MID$(F$(E),2,1)="p" AND N(E)=1 TH
EN G$="familia: boro"
580 IF MID$(F$(E),2,1)="p" AND N(E)=2 TH
EN G$="familia: carbono"
590 IF MID$(F$(E),2,1)="p" AND N(E)=3 TH
EN G$="familia: Nitrogenio"
600 IF MID$(F$(E),2,1)="p" AND N(E)=4 TH
EN G$="familia: Calcogenios"
610 IF MID$(F$(E),2,1)="p" AND N(E)=5 TH
EN G$="familia: Halogenios"
620 A$=MID$(F$(E),1,2):IF A$="4f"AND N(E
)<14THEN G$="familia: lantanidios"
630 IF A$="5f"AND N(E)<14THEN G$="famili
a: Actinidios"
640 IF MID$(F$(E),2,1)="d" AND N(E)<10TH
EN G$="-De transicao"
650 IF Z=70 OR Z=71  THEN G$="familia: l
antanidios"
660 IF Z=102 OR Z=103  THEN G$="familia:
 Actinidios"
670 IF VI(K)=8 THEN G$="Gases Nobres"
680 IF VI(K)=2 AND E$="HELIO" THEN G$="G
ases Nobres"
690 GOTO 1710
700 '
710 '   DISTRIBUICAO EM SUBNIVEIS
720 '     ENERGETICOS
730 '
740 COLOR 1,15:SCREEN2
750 KEY(1)ON:KEY(2) ON:KEY(3) ON:KEY(4)O
N:KEY(5) ON
760 CLOSE#1
770 OPEN"grp:"AS#1:XX=10:YY=80
780 LINE(5,50)-(250,180),1,BF
790 COLOR15
800 FOR X=1 TO E
810 PSET(XX,YY),1:PRINT#1,F$(X)
820 PSET(XX+7,YY-4),1:PRINT#1,N(X);
830 LINE(XX+28,YY+10)-(XX+40,YY-10)
840 XX=XX+40
850 IF XX>245 THEN XX=10:YY=YY+40
860 NEXT:COLOR1
870 LINE(5,5)-(250,35),7,BF
880 PSET(10,10),7:PRINT#1," Distribuicao
 em subniveis e-      nergeticos  de acord
o com o      Diagrama de Pauling(tecle F2
)"
890 GOTO 890
900 FOR X=1 TO K
910 PRINTVI(X):NEXT
920 FOR X=1 TO I:PRINTF$(X),N(X):NEXT
930 GOTO 1710
940 '
950 'DESENHA OS ELETRONS NAS CAMADAS
960 '
970 COLOR1,15:SCREEN2:C=20
980 KEY(1)ON:KEY(2)ON:KEY(3)ON:KEY(4)ON:
KEY(5)ON
990 R=RND(-TIME)
1000 FOR X=1 TO K
1010 P!=6.2856/VI(X)
1020 FOR J=0   TO 6.2856 STEPP!
1030 XC=128+C*SIN(J):YC=96+C*COS(J)
1040 CIRCLE(XC,YC),3,1
1050 PAINT(XC,YC),1
1060 NEXT
1070 CIRCLE(128,96),C,1
1080 C=C+12:NEXT
1090 CLOSE#1
1100 OPEN"GRP:" AS#1
1110 PRESET(10,180):COLOR 4
1120 PRINT#1,E$,"Z=";Z
1130 CIRCLE(128,96),7,1
1140 FOR X =1 TO 50
1150 A=RND(1)*6:B=RND(1)*6
1160 C=RND(1)*2:IF C=1 THEN B=-B
1170 D=RND(1)*2:IF D=1 THEN A=-A
1180 PSET(128+A,96+B),1
1190 NEXT
1200 GOTO 1200
1210 '
1220 'DIAGRAMA DE LINUS PAULING
1230 '
1240 COLOR 1,7,4:SCREEN2
1250 KEY(1)ON:KEY(2)ON:KEY(3)ON:KEY(4)ON
1260 CLOSE#1
1270 OPEN"GRP:" AS#1
1280 FOR I = 1 TO 7
1290 PRESET(50,20+20*I):I$=STR$(I)
1300 PRINT#1,I$+"s"
1310 NEXT
1320 FOR I = 2 TO 7
1330 PRESET(85,20+20*I):I$=STR$(I)
1340 PRINT#1,I$+"p"
1350 NEXT
1360 FOR I = 3 TO 7
1370 PRESET(120,20+20*I):I$=STR$(I)
1380 PRINT#1,I$+"d"
1390 NEXT
1400 FOR I = 4 TO 7
1410 PRESET(155,20+20*I):I$=STR$(I)
1420 PRINT#1,I$+"f"
1430 NEXT
1440 FOR I = 1 TO 7
1450 LINE(42,34+20*I)-(52,28+20*I)
1460 NEXT
1470 FOR I = 1 TO 8
1480 LINE(76,18+20*I)-(90,10+20*I):I$=ST
R$(I)
1490 NEXT
1500 FOR I = 2 TO 8
1510 LINE(76+35,18+20*I)-(90+35,10+20*I)
1520 NEXT
1530 FOR I = 3 TO 8
1540 LINE(76+70,18+20*I)-(90+70,10+20*I)
1550 NEXT
1560 FOR I = 4 TO 7
1570 LINE(181,18+20*I)-(195,10+20*I)
1580 NEXT
1590 FOR I=1 TO 7
1600 PRESET(42,34+20*I)
1610 LINE-(45,27+20*I)
1620 PRESET(42,34+20*I)
1630 LINE-(46,35+20*I)
1640 NEXT
1650 LINE(10,5)-(240,20),3,BF
1660 PRESET(3,10):PRINT#1,"     DIAGRAMA D
E LINUS PAULING"
1670 GOTO 1670
1680
```

```
1690 ' FICHA DOS ELEMENTOS
1700 '
1710 COLOR1,15,4:SCREEN2,1
1720 KEY(1)ON:KEY(2)ON:KEY(3)ON:KEY(4)ON
:KEY(5)ON
1730 LINE(30,30)-(220,160),7,BF
1740 X$=S$:GOSUB 2500
1750 PUT SPRITE1,(40,40),1,1
1760 PUT SPRITE2,(55,40),1,2
1770 CLOSE#1
1780 OPEN"GRP:"AS#1:C=7
1790 PSET(50,60),C:PRINT#1,"ELEMENTO:";E$
1600 PSET(57,72),C:PRINT#1,"-NUMERO ATOM
ICO:";Z
1810 PSET(57,82),C:PRINT#1,"-PESO ATOMIC
O:";M$
1820 NE=INT(VAL(M$))-Z
1830 PSET(57,92),C:PRINT#1,"-NEUTRONS:";
NE
1840 PSET(57,102),C:PRINT#1,"-CAMADAS:";K
1850 IF VI(K)<4 THEN V$="+"+STR$(VI(K))
1860 IF VI(K)>4 THEN V$="-"+STR$(8-VI(K))
1870 IF VI(K)=4 THEN PSET(57,112),C:PRIN
T#1,"-SEMICONDUTOR":GOTO 1890
1880 PSET(57,112),C:PRINT#1,"-VALENCIA:
";V$
1890 PSET(57,122),C:PRINT#1,G$
1900 GOTO 1900
1910 DATA HIDROGENIO,H,1,HELIO,He,4
1920 DATA LITIO,Li,6.94,BERILIO,Be,9.01
1930 DATA BORO,B,10.8,CARBONO,C,12
1940 DATA NITROGENIO,N,14,OXIGENIO,O,16
1950 DATA FLUOR,F,19,NEONIO,Ne,20.2
1960 DATA SODIO,Na,23,MAGNESIO,Mg,24.3
1970 DATA ALUMINIO,Al,27,SILICIO,Si,28.1
1980 DATA FOSFORO,P,31,ENXOFRE,S,32.1
1990 DATA CLORO,Cl,35.5,ARGONIO,Ar,39.9
2000 DATA POTASSIO,K,39.1,CALCIO,Ca,40.1
2010 DATA ESCANDIO,Sc,45,TITANIO,Ti,47.9
2020 DATA VANADIO,V,50.9,CROMIO,Cr,52
2030 DATA MANGANES,Mn,54.9,FERRO,Fe,55.8
2040 DATA COBALTO,Co,58.9,NIQUEL,Ni,58.7
2050 DATA COBRE,Cu,63.5,ZINCO,Zn,65.4
2060 DATA GALIO,Ga,69.7,GERMANIO,Ge,72.6
2070 DATA ARSENIO,As,74.9,SELENIO,Se,79
2080 DATA BROMO,Br,79.9,CRIPTONIO,Kr,83.8
2090 DATA RUBIDIO,Rb,85.5,ESTRONCIO,Sr,8
7.6
2100 DATA ITRIO,Y,88.9,ZIRCONIO,Zr,91.2
2110 DATA NIOBIO,Nb,92.9,MOLIBDENIO,Mo,9
5.9
2120 DATA TECNECIO,Tc,99,RUTENIO,Ru,101
2130 DATA RODIO,Rh,103,PALADIO,Pd,106
2140 DATA PRATA,Ag,108,CADMIO,Cd,112
2150 DATA INDIO,In,115,ESTANHO,Sn,119
2160 DATA ANTIMONIO,Sb,122,TELURIO,Te,128
2170 DATA IODO,I,127,XENONIO,Xe,131
2180 DATA CESIO,Cs,133,BARIO,Ba,137
2190 DATA LANTANIO,La,139,CERIO,Ce,140
2200 DATA PRASEODIMIO,Pr,141,NEODIMIO,Nd
,144
2210 DATA PROMECIO,Pm,147,SAMARIO,Sm,150
2220 DATA EUROPIO,Eu,152,GADOLINIO,Gd,157
2230 DATA TERBIO,Tb,159,DISPROSIO,Dy,163
2240 DATA HOLMIO,Ho,165,ERBIO,Er,167
2250 DATA TULIO,Tm,169,ITERBIO,Yb,173
2260 DATA LUTECIO,Lu,175,HAFNIO,Hf,178
2270 DATA TANTALIO,Ta,181,TUNGSTENIO,W,184
2280 DATA RENIO,Re,186,OSMIO,Os,190
2290 DATA IRIDIO,Ir,192,PLATINA,Pt,195
2300 DATA OURO,Au,197,MERCURIO,Hg,201
2310 DATA TALIO,Ti,204,CHUMBO,Pb,207
2320 DATA BISMUTO,Bi,209,POLONIO,Po,210
2330 DATA ASTATO,At,210,RADONIO,Rn,222
2340 DATA FRANCIO,Fr,223,RADIO,Ra,226
2350 DATA ACTINIO,Ac,227,TORIO,Th,232
2360 DATA PROTACTINIO,Pa,231,URANIO,U.238
2370 DATA NEPTUNIO,Np,237,PLUTONIO,Pu,242
2380 DATA AMERICIO,Am,243,CURIO,Cm,247
2390 DATA BERQUELIO,Bk,247,CALIFORNIO,Cf
,251
2400 DATA EINSTEINIO,Es,254,FERMIO,Fm,253
2410 DATA MENDELEVIO,Md,256,NOBELIO,No,2
53
2420 DATA LAWRENCIO,Lr,257
2430 DATA 1s,2s,2p,3s,3p,4s,3d,4p,5s,4d,
5p,6s.4f,5d,6p,7s,5f,6d,7p
2440 '
2450 '      CRIA SPRITES AMPLIADOS COM
2460 '      SIMBOLOS DOS ELEMENTOS
2470 '
2480 X$=S$
2490 SCREEN1,1
2500 DEF FNS=PEEK(A*8+&H1BBF+S)
2510 FOR L=1 TO LEN(X$)
2520 A=ASC(MID$(X$,L,1))
2530 FOR S=0 TO 7
2540 Q$=Q$+CHR$(FNS)
2550 NEXT
2560 SPRITE$(L)=Q$:Q$=""
2570 NEXT
2580 RETURN
```

*SÉRGIO DURIC CALHEIROS*

# MSX DEBUG

## PARTE 3

No mês passado, vimos como o MSXDEBUG reconhecia os comandos e chamava suas respectivas rotinas. Além disso, deixamos o caminho preparado para implementação do comando SOMA. Todos os comandos, excetuando o comando DOS, precisam de um ou mais parâmetros. Neste mês, veremos como essas rotinas manipulam estes parâmetros. Para entender este mecanismo, tomaremos como exemplo a rotina @SOMAR que será implementada no MSXDEBUG.

Como já foi dito, tudo que é digitado fica armazenado num buffer de memória. A primeira ação do MSXDEBUG, é, então, passar o controle para a rotina de reconhecimento de comandos @INSTR. Se o comando não existir, será enviada uma mensagem de erro e o conteúdo anterior do buffer se perderá. Caso contrário, o controle será passado para a rotina que corresponde ao comando dado. No buffer restam os parâmetros digitados, se houver algum. O comando terá sido eliminado pela rotina @INSTR.

A partir deste ponto, cabe à rotina chamada por @INSTR reconhecer e trabalhar com os parâmetros do buffer. Podemos entender como isso acontece, examinando a rotina @SOMAR. Na listagem 2, estão comentadas as rotinas envolvidas no processo. Antes de um exame detalhado, convém descrever qual foi a lógica usada na rotina @SOMAR.

Os parâmetros que @SOMAR precisa, são os endereços inicial e final do bloco de memória. Existem rotinas especializadas em transformar os parâmetros que estão no buffer para uma forma em que estes valores possam ser manipulados como números.

A rotina @GTDAT se encarrega de encontrar o primeiro parâmetro do buffer. A rotina @GTD07 organiza os parâmetros. Finalmente, temos a rotina @CONVD, que se encarrega de converter o dado para seu valor numérico. Fazendo analogia com o BASIC, a rotina @CONVD pode ser comparada com a função VAL, onde o registrador HL aponta a STRING e, na saída, devolve o valor já convertido. Usando esses recursos, qualquer outra rotina pode fazer uso dos parâmetros que lhes forem repassados.

De posse dos parâmetros que definem os limites do bloco, resta, então, somar sucessivamente os bytes que estiverem lá. Entretanto, existe um porém. Sabemos que, em linguagem de baixo nível, o valor máximo que pode ser manipulado é um número de 16 bits. Este número é, com certeza, insuficiente para armazenar a soma de um bloco, mesmo que seja pequeno. Para isso, então, basta construir um número de 32 bits. Isso pode ser facilmente conseguido com a manipulação simultânea de dois números de 16 bits. Foram usados os registradores de index IX e IY do Z-80. O registrador IX serve para acumular os dados e IY para acumular algum eventual estouro do registrador anterior. Como isso não acontece automaticamente, deve-se verificar o estado da flag de CARRY após cada soma. Ao final do loop, teremos o valor da soma com a parte mais significativa em IY e a menos significativa em IX. Como último procedimento, resta imprimir o número, que deve ser feito em partes (não há outra maneira). Assim, temos um número de 32 bits que, certamente, comporta o volume de qualquer soma.

Agora, estamos prontos para implementar o comando SOMA. Como de praxe, carregue o MSXDEBUG no endereço 4100H. A partir do endereço 4E46H, digite o bloco de listagem 1 com o comando DISP. Isto inclui deletar o BYTE 0C9H que existia no endereço 4E46H.

Se quiser atualizar a versão do MSXDEBUG, procure a mensagem inicial no final do programa e faça a devida correção. Uma sugestão: para que possamos reconhecer que esta é a versão com o comando SOMA, troque a mensagem Vr 1.0 para Vr 1.1.

Para testar a implementação, saia do MSXDEBUG antigo e execute a nova versão. Além da me sagem, não deve ser notada nenhuma mudança. Digite então: SOMA 100 0B7A (CR). A soma do bloco deverá ser mostrada em hexadecimal. Deve ser igual a 3CEE9.

A partir deste número, utilizaremos, para quem implementou o comando SOMA, mais um recurso que prevenirá contra erros de digitação.

Os leitores que já se aventuraram pelo mundo do assembler, já devem ser capazes de implementar suas próprias rotinas no MSXDEBUG. Aqueles que estão iniciando não devem deixar passar as dicas e as técnicas de programação nesta linguagem. O MSXDEBUG não é apenas mais um programa; é também um curso de aperfeiçoamento!

Até a próxima! ■

**LISTAGEM 1**

```
4E40 00 00 00 00 00 00 CD C9
4E48 08 DD 21 00 00 FD 21 00
4E50 00 ED 5B 8B 0D 2A 89 0D
4E58 13 4E 06 00 DD 09 30 02
4E60 FD 23 23 CD 33 0B 20 F1
4E68 FD 22 85 0D 11 B6 0D D5
4E70 CD 09 08 AF 12 D1 CD 24
4E78 0B DD 22 85 0D C3 E7 07
```

OBSERVAÇÕES:

1) A rotina que funciona como a função STR$ do BASIC é a rotina @RECON (no endereço 07E7H). Para utilizá-la, devemos colocar o valor de 16 bits no endereço RECDT (0D85H). O número será impresso na posição corrente da tela. Entretanto, esta posição não será alterada, ou seja, a próxima impressão será feita no mesmo lugar do número impresso.

2) As rotinas @GTDAT, @GTD07 e @CONVD localizam-se nos endereços 089AH, 08C9H e 08FAH, respectivamente

**LISTAGEM 2**

```
GTDAT:  CALL CON02        ; Verifica buffer e aponta dado
        AND A
        RET NZ            ; Se não estiver vazio, tudo bem.
        LD DE,ERR02       ; Caso contrário, erro!
        JP ERROR

GTD07:  CALL GTDAT        ; Acha parâmetro no buffer
        CALL CONVD        ; Converte seu valor e o elimina do buffer
        LD (ENDIN),HL     ; Armazena na memória
        CALL GTDAT        ; Repete seqüência para outro dado
        CALL CONVD
        LD (ENDFI),HL
        LD DE,(ENDIN)     ; Verifica se dados são coerentes
        CALL CMPRG        ; Compara DE e HL (início e fim)
        RET NC            ; Se início <= fim, tudo bem
        LD DE,ERR03       ; Caso contrário, erro!
        JP ERROR

SOMAR:  CALL GTD07        ; Organiza os parâmetros
        LD IX,0           ; Zera contadores
        LD IY,0
        LD DE,(ENDFI)     ; Recupera dados fornecidos no buffer
        LD HL,(ENDIN)
        INC DE
SOM01:  LD C,(HL)         ; Inicia loop de soma
        LD B,0
        ADD IX,BC         ; Acumula dado
        JR NC,SOM02       ; Soma CARRY se ocorrer
        INC IY
SOM02:  INC HL
        CALL CMPRG        ; Compara limites do bloco
        JR NZ,SOM01       ; Repete se não terminou
        LD (RECDT),IY     ; Prepara valores para impressão
        LD DE,RECBF
        PUSH DE
        CALL REC03        ; Reconverte número para STRING
        XOR A
        LD (DE),A         ; Prepara impressão do número.
        POP DE
        CALL MSOUT        ; Imprime parte do número de 32 bits
        LD (RECDT),IX
        JP RECON          ; Imprime resto do número de 32 bits
```

# TRAFALGAR PLUS

*FERNANDO LEIBEL*

Trafalgar Plus é uma nova versão do jogo do tipo wargame para duas pessoas. Para quem não conhece o jogo, a missão dos jogadores é a de comandar um complexo militar que é formado por infantaria, cavalaria e também por uma esquadra, sendo composto por um tabuleiro, que se encontra dividido em duas partes, a terrestre e a marítima.

Esta nova versão conta com um editor que possibilita ao usuário criar o tabuleiro em que vai jogar. Você pode redefinir todos os caracteres que fazem parte do jogo, tanto na forma quanto nas cores. Desta maneira, o cavalo do Trafalgar original pode se transformar no avião a jato do seu jogo.

Os caracteres, mesmo com a forma e a cor alterados, continuam desempenhando as mesmas funções, ou seja, se redefinirmos um caracter de montanha identicamente a um caracter do mar, apesar da aparência, ele continuará a funcionar como um caracter de montanha.

No Trafalgar, cada exército era constituído por 22 peças. No seu jogo, você é quem define quantas peças constituirão um exército. A forma de se jogar continuar sendo a mesma.

**Editor**

O Editor tem como função permitir a criação do tabuleiro pelo usuário.

Menu principal:

A) Compor gráfico
B) Redefinir caracteres
C) Colorir gráfico
D) Gravar gráfico
E) Buscar gráfico
F) Limpar gráfico
G) Fim

**Compor gráfico**

Nesta opção, você possui um cursor que pode ser movimentado com as setas.

Com as teclas "A" e "S", você escolhe o caracter que será impresso.

RETURN imprime o caracter.

CONTROL + STOP retorna ao menu principal.

**Redefinir caracteres**

Com esta opção, você muda a forma dos caracteres.

As teclas de "A" a "S" selecionam o caracter a ser redefinido.

Com a barra espaçadora, setamos ou resetamos o modelo do caracter em tamanho ampliado.

As setas movimentam o cursor do modelo ampliado.

RETURN copia a forma do modelo ampliado para o caracter selecionado.

**Colorir gráfico**

Na opção de colorir gráficos, podemos trocar as cores dos caracteres, obedecendo os grupos a que eles pertencem.

O grupo que vai ser alterado deve ser selecionado com as setas do cursor.

Com as teclas "P" e "S" seleciona-se a cor de frente e a cor de fundo.

RETURN marca a cor selecionada.

**Gravar gráfico**

Esta opção grava o tabuleiro que foi criado, em disco ou cassete.

**Buscar gráfico**

Lê o gráfico do disco ou do cassete.

**Limpar gráfico**

Limpa o tabuleiro que está sendo elaborado. As cores selecionadas e as redefinições efetuadas continuam.

## Fim

Abandona e retorna ao Basic.

## Regras

Algumas regras devem ser seguidas ao se montar um tabuleiro.

– Os barcos só andam em mar aberto, que é caracterizado por um caracter cheio, em cor azul.

– Os portos devem ficar acima do mar aberto.

– Os guardas reais só se locomovem dentro dos estados.

– Só pode haver um navio cargueiro para cada lado.

## Trafalgar – Instruções

Trafalgar é um jogo do tipo war-game, disputado por duas pessoas, cada uma comandando um exército.

## Descrição do tabuleiro e peças

O tabuleiro é composto por 5 partes principais. Suas descrições e cores são as seguintes:

– Território Francês (azul/branco)
– Território Inglês (vermelho/branco)
– Zona neutra de combate (preto/branco)
– Mar (azul)
– Montanhas (magenta)
– Ilhas (magenta)
– Árvores (verde)
– Portos de Embarque
– Recifes (localizados na zona de contato entre o mar e a parte terrestre, exceto nos portos)
– Castelos (vermelho ou azul). Cada exército possui um castelo, que se localiza em seu território.

Cada exército conta, de início, com 22 peças, identificadas na figura 2. Elas se movimentam da seguinte maneira:

– Na sua vez, o jogador poderá mover qualquer peça do seu exército, movimentando o cursor com as setas, colocando-o sobre a peça a ser movida, digitando ENTER e repetindo o mesmo procedimento para a posição de destino dessa peça.

– As peças se movimentam de uma em uma casa, à exceção da cavalaria, que pode andar de duas em duas casas.

– Não se pode mover as peças da

```
1 CLEAR700,&HDFFF
2 DEFINTA-Z
3 ONERROR GOTO 20000
5 CA$="033128136040004103403503603703803904804905005605712012112212312412509609709809910010110210310410510610710811211311411511611913012914414514614714815115315 2":CA =1
6 DIMRE(8,8),CO(16)
7 DATA239,206,239,94,30,30,30,30,110,110,244,94,79,143,248,143
8 FORT=1TO16:READA:CO(T)=A :NEXT
10 SCREEN1,1:KEYOFF
11 GOSUB5500
15 GOSUB510
17 GOSUB5000
18 ONSTOPGOSUB5400
19 STOPON
20 ONINTERVAL=10GOSUB5310
25 CLS:INTERVALOFF:CR=0:GOSUB5300
30 LOCATE 0,3:PRINT          "*******EDITOR DE TELAS*******"
35 LOCATE0,7
40 PRINT"A]COMPOR GRAFICO"
50 PRINT"B]REDEFINIR CARACTERES          C]COLORIR GRAFICO"
60 PRINT"D]GRAVAR GRAFICO"
70 PRINT"E]BUSCAR GRAFICO"
80 PRINT"F]LIMPAR GRAFICO          G]FIM"
90 LOCATE0,17:          PRINT"****************************"
100  M$=INKEY$
110 IFM$<>""THENM=ASC(M$):GOTO130
120 GOTO100
130 IF M>64ANDM<72THENONM-64GOTO650,1000,1500,2300,2500,3000,3500
140 GOTO100
510  DATA 48,08,10,08,10,08,10,08,00, 49,FE,FF,FE,FF,FF,FE,FF,5A, 50,FF,FF,FF,FF,FF,FF,FF,5A, 119,24,18,3C,3C,18,18,18,7E, 151,24,18,3C,3C,18,18,18,7E, 33,FF,81,81,81,81,81,81,FF, 128,FF,81,81,81,81,81,81,FF, 136,FF,81,81,81,81,81,81,FF
520  DATA 40,08,1C,3F,7E,7F,FE,FC,38, 96,FF,7F,3F,1F,3F,1F,3F,4B, 97,2C,7E,FF,7F,FE,7C,7E,FF, 98,FF,FE,7E,7F,FE,7C,FE,FF, 99,FF,FF,7F,3F,3F,7F,FF,FF, 124,FF,FF,FF,FF,FF,FF,FF,FE, 125,D4,CA,FC,FF,FC,FE,FF,FF, 32,2D,FF,7F,FF,FF,7F,FF,7F
530  DATA 34,7F,FF,7F,FF,FF,7F,FF,7F, 35,FF,E7,C3,81,81,C3,E7,FF, 116,38,10,10,38,7C,7C,18,7E, 148,38,10,10,38,7C,7C,18,7E, 41,1F,18,18,1F,18,1F,18,1F, 112,04,24,24,24,24,2E,FF,7E, 100,FF,FF,FF,FF,FF,FF,FF,FF, 101,FF,FF,FF,FF,FF,FF,FE,95
540  DATA 102,54,FE,FE,FE,FF,FC,FC,AA, 103,D4,CA,FC,FF,FC,FE,FF,FF, 104,44,EF,FF,FF,FF,FF,FF,FF, 105,88,DD,FF,FF,FF,3F,7F,92, 106,FF,FE,FF,FC,F8,FF,FC,FF, 107,FF,FF,FC,FF,FC,FE,FF,D5, 108,3F,7F,3F,FF,3F,EF,FF,53, 56,FF,7C,FF,FC,3F,FF,7F,A9
550  DATA 57,01,51,D5,7F,FF,3F,7F,FF, 120,FF,FF,FF,FF,FF,FF,FF,FF, 121,FF,FE,FF,FF,F8,FF,FC,FF, 122,44,EF,FF,FF,FF,FF,FF,FF, 123,FF,FF,FC,FF,FC,FE,FF,D5, 36,FF,FF,FF,FF,FF,FF,FF,FF, 37,5A,FF,FF,FF,FF,FF,FF,FF, 38,00,00,00,AA,55,00,00,00
560 DATA39,FE,FF,FE,FF,FF,FE,FF,FE,129,00,0C,1E,3E,7A,78,38,7E,152,00,0C,1E,3E,7A,78,38,7E,130,3C,18,18,18,BD,FF,99,18,153,3C,18,18,18,BD,FF,99,18,144,20,24,24,24,24,74,FF,7E,113,04,24,24,24,24,2E,FF,7E,114,04,24,24,24,24,2E,FF,7E,115,04,24,24,24,24,2E,FF
570 DATA7E, 145,20,24,24,24,24,74,FF,7E,146,20,24,24,24,24,74,FF,7E,147,20,24,24,24,24,74,FF,7E,2,81,81,81,81,81,81,81,81,3,FF,00,00,00,00,00,00,FF
575 RESTORE510
580  FORT=1TO54:READA:FORS=0TO7:READB$:VPOKEB*A+S,VAL("&H"+B$):NEXTS:NEXTT
590 VPOKE 8192,&H99
620 RETURN
650 CM=1:CLS:FORT=1TO16:VPOKE8195+T,CO(T):NEXT
651 IFCP=1GOTO5600
652 LOCATE0,2: FORT=1TO23: PRINT" !!!!!!!!!!!!!!!!!!!!!!!!!!!!!      ";:NEXT
655 D$="!":E$=" ":FORT=1TO8:D$=D$+CHR$(128):E$=E$+CHR$(136):NEXT
660 FORT=0TO7:LOCATE0,T:PRINT E$;:NEXT
670 FORT=15TO22:LOCATE18,T:PRINTD$;:NEXT
671 CP=1
675 LOCATE2,23:PRINT"CARACTER] ! [";:CB=33
676 INTERVALON
677 R=0:R1=0
690 K$=INKEY$
700 IFK$<>""THENK=ASC(K$):GOTO720
710 GOTO690
720 IFK>27ANDK<32THENGOSUB5160:GOTO690
730 M=INT((Y1+9)/8):N=INT((X1+9)/8)
740 IF K=83THEN CB =VAL(MID$(CA$,CA ,3)):VPOKE6894,CB :CA =CA +3:IFCA>153THENCA=1
745 IF K=65THEN CB =VAL(MID$(CA$,CA ,3)):VPOKE6894,CB :CA =CA -3:IFCA<1  THENCA=154
750 IFK=13THEN VPOKE(6143+32*(M-1)+N),CB
755 GOTO690
900 GOTO900
1000 CLS:X1=95:Y1=47:GOSUB5300
1002 FORT=1TO16:VPOKE8195+T,CO(T):NEXT
1005 FORT=1TO8:FORT1=1TO8:RE(T,T1)=0:NEXT:NEXT
1010 FORT=1TO8:LOCATE10,5+T:PRINTSTRING$(8,128):NEXT
1020 LOCATE2,23:PRINT"CARACTER] ! [";:CB=33
1023 INTERVALON
1025 R=72:R1=48
1030 K$=INKEY$
1040 IFK$<>""THENK=ASC(K$):GOTO1060
1050 GOTO1030
1060 IFK>27ANDK<32THENGOSUB5160:GOTO1030
1070 IF K=83THEN CB =VAL(MID$(CA$,CA ,3)):VPOKE6894,CB :CA =CA +3:IFCA>147THENCA=1
1080 IF K=65THEN CB =VAL(MID$(CA$,CA ,3)):VPOKE6894,CB :CA =CA -3:IFCA<1  THENCA=148
1180 M=INT((Y1+9)/8):N=INT((X1+9)/8)
1190 IFK=13GOTO1300
1200 IFK=32GOTO1250
1205 IFK=76GOTO1000
1210 GOTO1030
1250 IFRE(M-6,N-12)=0 THENVPOKE(6143+32*(M-1)+N),219:RE(M-6,N-12)=1:GOTO1030
1260 VPOKE(6143+32*(M-1)+N),128:RE(M-6,N-12)=0:GOTO1030
1300 INTERVALOFF:CR=0:GOSUB5300
1310 FORT=1TO8:RE$="":FORT1=1TO8:R1$=STR$(RE(T,T1)):RE$=RE$+RIGHT$(R1$,1):NEXT
1320 VPOKE8*CB+T-1,VAL("&B"+RE$):NEXT
1325 INTERVALON
1330 GOTO1030
1500 CLS
1505 VPOKE8220,&H11:VPOKE8221,&HFF:CO=1:CF=15
1510 X=0:Y=6:LOCATE0,6:PRINT" ]AREA CINZA/CASTELO"
1520 PRINT" ]ARVORE/PORTO"
1530 PRINT" ]MAR"
1540 PRINT" ]MONTANHA"
1550 PRINT" ]EXERCITO AZUL"
1560 PRINT" ]EXER. AZUL/AREA AZUL"
1570 PRINT" ]AREA VERMELHA"
1580 PRINT" ]EXERCITO VERMELHO"
1590 PRINT" ]EXERCITO VERMELHO"
1595 PRINT:PRINT"COR DE FRENTE ";CHR$(224)
1596 PRINT"COR DE FUNDO  ";CHR$(232)
1597 REM
1598 LOCATE X,Y:PRINT">":PLAY"BEG"
1600 M$=INKEY$:IF M$="" GOTO 1600
1602 IF ASC(M$)=13 THEN LOCATE X,Y:PRINT" ":GOSUB 2100:GOTO 1630
1605 IF ASC(M$)=30 THEN LOCATE X,Y:PRINT" ":Y=Y-1
1607 IF ASC(M$)=31 THEN LOCATE X,Y:PRINT" ":Y=Y+1
1610 IF Y<6 THEN Y=14
1615 IF Y>14 THEN Y=6
1617 LOCATE X,Y:PRINT">"
1620 GOTO1600
1630 ONY-5 GOTO1650,1700,1750,1800,1850,1900,1950,2000,2050
1650 G=1:GOSUB2200:G=3:GOSUB2200:GOTO1597
1700 G=2:GOSUB2200:GOTO1597
1750 G=4:GOSUB2200:G=12:GOSUB2200:GOTO1597
1800 G=9:GOSUB2200:G=10:GOSUB2200:GOTO1597
1850 G=11:GOSUB2200:GOTO1597
1900 G=13:GOSUB2200:GOTO1597
1950 G=14:GOSUB2200:GOTO1597
2000 G=15:GOSUB2200:GOTO1597
2050 G=16:GOSUB2200:GOTO1597
2100 PLAY"GEB"
2105 M$=INKEY$:IFM$<>""THENM=ASC(M$):GOTO2120
2110 GOTO2105
2120 IFM=65THENCO=CO+1:IFCO=16THENCO=1
2130 IFM=83THENCF=CF+1:IFCF=16THENCF=1
2140 IFM=13THENCO$=HEX$(CO)+HEX$(CF):RETURN
2150 CO$=HEX$(CO)+HEX$(CO)
2160 CF$=HEX$(CF)+HEX$(CF)
2170 VPOKE8220,VAL("&H"+CO$):VPOKE8221,VAL("&H"+CF$)
2180 GOTO2105
2200 CO(G)=VAL("&H"+CO$):RETURN
2300 LOCATE0,20:PRINT" ]AGUARDE UM MOMENTO"
2310 A$="L32GEB"
2320 PLAYA$
2330 FORT=0TO2048:B=VPEEK(T):POKE40732!+T,B:NEXT
2340 FORT=0TO15:POKE42781!+T,CO(T+1):NEXT
```

zona neutra, nem para as montanhas nem para o mar. As peças também não podem passar por árvores, com exceção de cavalaria.

– Os navios não se movimentam junto aos recifes.

– Os Guarda Reais não saem de seu território.

– As peças não podem se movimentar na diagonal.

**O Jogo**

– O jogo é sempre iniciado pelo lado vermelho (França).

– Cada jogador terá direito a um número, seguido de lances, que varia de 5 a 7, dependendo de um sorteio feito pelo computador. Toques de sino indicam que é chegada a vez do outro jogador.

– São considerados lances:

* O movimento de uma peça.

* Ataque (terrestre ou naval).

As operações de embarque e desembarque, assim como os lances incorretos, não serão computados.

**Embarque**

Cada exército possui apenas um navio capaz de embarcar e transportar peças. Cada um desses navios cargueiros podem carregar até 3 peças, independente do tipo. O embarque pode ser total (3 peças) ou parcial (1 ou mais peças) em cada porto.

O procedimento do embarque é o seguinte:

* Posicione as peças a serem embarcadas nas casas imediatamente acima, à esquerda e à direita do porto onde o navio está atracado.

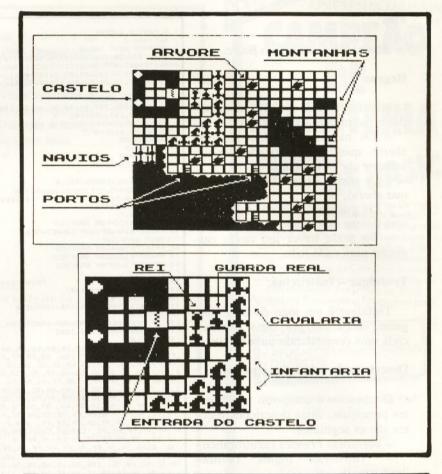
* Pressione a tecla 1.

* Coloque o cursor sobre o cargueiro e tecle ENTER.

O navio cargueiro, por motivos estratégicos, não difere dos demais. Para saber qual é o navio cargueiro de seu exército, pressione a tecla 6.

O navio cargueiro pode, ainda, seqüestrar peças inimigas que estejam nas posições de embarque no momento do mesmo.

**Desembarque**

O procedimento é o mesmo do embarque, com a diferença que se deve teclar 1 em vez de 2. No momento do desembarque, as peças que por acaso estiverem nas posições de embarque serão eliminadas.

```
2350 LOCATE0,20:PRINTSTRING$(32,32):LOCA
TE0,20::LINE INPUT"DE O NOME ";TT$:LOCAT
E 0,20:PRINT" ]PREPARE PARA GRAVACAO   "
:PLAYA$
2360 B$=INKEY$:IFB$=""THEN2360
2370 LOCATE0,20:PRINT" ]GRAVANDO
             "
2375 BSAVE TT$,40000!,42796!
2380 LOCATE0,20:PRINT" ]GRAVADO
  ":PLAYA$:FORT=1TO3000:NEXT:GOTO25
2500 A$="L32GEB"
2520 LOCATE0,20:PRINTSTRING$(32,32):LOCA
TE0,20:LINE INPUT "DE O NOME ";TT$:LOCAT
E0,20:PRINT" ]PREPARE PARA LEITURA":PLAY
A$
2530 B$=INKEY$:IFB$=""THEN2530
2560 LOCATE0,20:PRINT" ]LENDO
          "
2565 BLOAD TT$
2570 LOCATE0,20:PRINT" ]AGUARDE UM MOMEN
TO     ":PLAYA$
2580 FORT=0TO2048:B=PEEK(40732!+T):VPOKE
T,B:NEXT
2590 FORT=0TO15:B=PEEK(42781!+T):CO(T+1)
=B:NEXT
2600 CP=1:GOTO25
3000 LOCATE0,20:PRINT" ]COMFIRMA S/N
       ":PLAY"L32AGE"
3005 B$=INKEY$:IFB$=""THEN3005
3010 IFB$="S"THENCP=0
3020 GOTO25
```

### Ataques

a) Terrestre: Para eliminar uma peça inimiga, é necessário atacá-la com forças mais poderosas. É importante notar que as casas diagonais à casa da peça atacada não contam na hora do confronto. Assim, as peças atacantes devem ser posicionadas nas casas imediatamente acima, abaixo, à esquerda e à direita da peça atacada. O mínimo para eliminar cada peça é:

* Rei – Qualquer peça.
* Guarda Real – Apenas o REI.
* Cavalaria – 1 Inf., 2 Cav. ou 1 Guarda Real.
* Infantaria – 2 Cav., 2 Inf., 1 Cav. e 1 Inf. ou 1 Guarda Real.

Para atacar, pressione a tecla 3, posicione o cursor sobre a peça a ser atacada e tecle ENTER. Se o ataque for vitorioso, a peça eliminada irá sumir do tabuleiro.

b) Naval: Os navios podem apontar seus canhões para esquerda ou direita, tendo um alcance máximo de 3 casas. Quanto mais perto estiver o navio atacante do navio atacado, maior será a probabilidade do ataque ser bem sucedido.

Para atacar, coloque o cursor sobre o navio atacante e digite 4 ou 5 (canhão para esquerda ou direita, respectivamente).

Observação: Se o navio afundado for um cargueiro, as peças que ele estiver transportando serão dadas como desaparecidas.

O jogo termina quando um dos exércitos consegue levar seu Rei até o castelo adversário. O Rei deverá entrar pela porta do castelo ou eliminar o Rei inimigo.

O desaparecimento do Rei, no caso dele estar a bordo de um cargueiro afundado, não determina o fim do jogo.

### Gravação

Como cada confronto pode durar horas ou até mesmo dias, o programa oferece a opção de gravação do jogo para posterior continuação.

Tanto o Editor como o Trafalgar foram escritos totalmente em Basic, para facilitar o entendimento dos usuários.

Na hora de carregar o Trafalgar, resete o micro. ■

```
3500 LOCATE0,20:PRINT" ]COMFIRMA S/N
     ":PLAY"L32AGE"
3505 B$=INKEY$:IFB$=""THEN3505
3507 IFB$="S"THEN3520
3510 GOTO25
3520 SCREEN0:END
5000 REM DEF SPRITE CURSOR
5010 FORT=1TO8
5020 READ A$
5030 S$=S$+CHR$(VAL("&B"+A$))
5040 NEXTT
5050 SPRITE$(1)=S$
5060 X1=127:Y1=95
5070 INTERVALON: RETURN
5080 DATA 11111000
5090 DATA 10001000
5100 DATA 10001000
5110 DATA 10001000
5120 DATA 11111000
5130 DATA 00000000
5140 DATA 00000000
5150 DATA 00000000
5160 REM SUB-ROTINA CURSOR
5170 C1=STICK(0):PLAY"V6L64N40"
5180 IFC1=1THENY1=Y1-8
5190 IFC1=2THENY1=Y1-8:X1=X1+8
5200 IFC1=3THENX1=X1+8
5210 IFC1=4THENY1=Y1+8:X1=X1+8
5220 IFC1=5THENY1=Y1+8
5230 IFC1=6THENY1=Y1+8:X1=X1-8
5240 IFC1=7THENX1=X1-8
5250 IFC1=8THENY1=Y1-8:X1=X1-8
5260 IFX1<23+RTHENX1=223-R
5270 IFX1>223-RTHENX1=23+R
5280 IFY1<-1+R1THENY1=175-R
5290 IFY1>175-RTHENY1=-1+R1
5300 PUT SPRITE0,(X1,Y1),CR,1:RETURN
5310 CR=CR+1:IFCR=16THENCR=0
5320 GOTO 5300
5400 IFCM=0GOTO5410
5401 INTERVALOFF:CR=0:GOSUB5300
5405 FORT=0TO731:B=VPEEK(6146+T):POKE400
00!+T,B:NEXT:CM=0
5410 CLS:INTERVALON:GOSUB5500:RETURN25
5500 RESTORE5510:FORT=1TO6:READA$,B$:VPO
KE8195+VAL(A$)    ,VAL(B$):NEXT
5510 DATA 1,68,2,244,5,244,6,244,7,244,8
,244
5520 RETURN
5600 FORT=0TO731:B=PEEK(40000!+T):VPOKE6
146+T,B:NEXT:GOTO675
20000 PLAY"AGA":RESUME 25
```

```
1 SCREEN 0
10 KEYOFF
20 DIMYF(7):CLEAR3000,47000!:MAXFILES=2
30 DATA 0,0,262,196,220,165,175,131,175,
196,65,0,0
40 READ N,V      ,YF(0),YF(1),YF(2),YF(3)
,YF(4),YF(5),YF(6),YF(7),L,CW,KOY
50 DIMNAV(2,3)
60 CLS:ON INTERVAL=10GOSUB4670
70 GOTO 4730
80 GOSUB4830
90 IF U$="V"OR U$="v"THENME$=">>LENDO":G
OTO3670
100 KEYOFF:SCREEN1,1,0
110 PLAYFR$
120 GOSUB10000
360 IF U$="V"OR U$="v"THENGOSUB4370:GOTO
3690
430 GOSUB4370
440 REM
450 JOG=5+INT(RND(-TIME)*3)
460 CW=CW+1
470 IF CW/2<>INT(CW/2)THENVPOKE8192,&H29
 ELSE VPOKE8192,&HF4
480 BEEP: FORT=1TO3: SOUND8,&B00010000:S
OUND9,&B00010000:SOUND 2,&B00010000
490 SOUND11,50:SOUND12,100:SOUND13,0
500 FORS=1TO800:NEXT:NEXT
510 K$=INKEY$
520 IFK$<>""THENK=ASC(K$):GOTO540
530 GOTO510
540 IF K=13THENM=INT((Y1+9)/8):N=INT((X1
+9)/8):D=33:GOSUB2960:GOTO580
550 IF K>47ANDK<55THENGOSUB2960:GOTO4290
560 IFK>27ANDK<32THENGOSUB4530:GOTO510
570 GOTO 2620
580 T=VPEEK(6143+32*(M-1)+N)
590 IFCW/2<>INT(CW/2)THEN640
600 IFT=129THEN790
610 IFT>111ANDT<116THEN950
620 IFT=116ORT=119ORT=130THEN680
630 GOTO2620
640 IFT=153ORT=148ORT=151THEN680
650 IFT=152THEN790
660 IFT>143ANDT<148THEN950
670 GOTO2620
680 GOSUB2980
690 C=1
700 GOTO2270
710 IFT=116ORT=148THENGOTO4800
720 IFM=ITHEN750
730 IFN=JTHEN780
740 GOTO2620
750 IFJ1<>1THEN740
760 GOSUB2740
770 GOTO 3640
780 IFI1<>1THEN740ELSE760
790 C=2
800 F=1
810 GOSUB2980
820 GOTO2270
830 IFJ1=1ORI1=1THEN850
840 GOTO2350
850 IFM=ITHEN880
860 IFN=JTHEN920
870 GOTO2620
880 IFJ1=1THEN900
890 IFJ1<>2THEN2620
900 GOSUB2740
910 GOTO3640
920 IFI1=1THEN900
930 IFI1<>2THEN2620ELSE900
940 C=4
950 GOSUB2980
960 GOTO2500
970 IFM=ITHEN1000
980 IFN=JTHEN1030
990 GOTO2620
1000 IFJ1<>1THEN990
1010 D=120
1020 GOTO3640
1030 IFI1<>1THEN2620ELSE1010
1040 GOSUB2980
1050 PAR=VPEEK(6143+32*(I-1)+J)
1060 IFPAR=113ORPAR=145THEN1080
1070 GOTO2620
1080 HGI=VPEEK(6143+32*(I-2)+J)
1090 TER=0
1100 IFHGI=41THEN1120
1110 GOTO2620
1120 P=VPEEK(6143+32*(I-3)+J)
1130 C=1
1140 IFP=33THEN1160
1150 GOTO1310
1160 P=VPEEK(6143+32*(I-2)+J+1)
1170 C=2
1180 IFP=33THEN1200
1190 GOTO1310
1200 P=VPEEK(6143+32*(I-2)+J-1)
1210 C=3
1220 IFP=33THEN3510
1230 GOTO1310
1240 GOTO2620
1250 VPOKE6143+32*(I-3)+J,33
1260 'SOM DE EMBARQUE
1270 PLAY"GEB"
1280 VPOKE6143+32*(I-2)+J+1,33
1290 VPOKE6143+32*(I-2)+J-1,33
1300 GOTO510
1310 TER=2
1320 IFPAR=113THEN1350
1330 R=1
1340 GOTO1370
1350 R=2
1360 GOTO1370
1370 NAV(R,C)=P
1380 ONCGOTO 1160,1200,1250
1390 GOSUB2980
1400 P=VPEEK (6143+32*(I-2)+J)
1410 IF P<>41THEN2620
1420 PAR=VPEEK(6143+32*(I-1)+J)
1430 IF PAR=113THEN1480
1440 IFPAR=145THEN1460
1450 GOTO2620
1460 R=1
1470 GOTO1490
1480 R=2
1490 IFNAV(R,1)+NAV(R,2)+NAV(R,3)=0THEN2
620
1500 GOSUB3520
1510 'SOM DE EMBARQUE
1520 PLAY"GEB"
1530 VPOKE6143+32*(I-3)+J,NAV(R,1)
1540 VPOKE6143+32*(I-2)+J+1,NAV(R,2)
1550 VPOKE6143+32*(I-2)+J-1,NAV(R,3)
1560 NAV(R,1)=0
1570 NAV(R,2)=0
1580 NAV(R,3)=0
1590 GOTO510
1600 GOSUB2980
1610 P=VPEEK(6143+32*(I-1)+J)
1620 IFP=130THEN1710
1630 IFP=153THEN1880
1640 IFP=129THEN2050
1650 IFP=152THEN2160
1660 IFP=116THEN3420
1670 IFP=148THEN3350
1680 IFP=119THEN3140
1690 IFP=151THEN3210
1700 GOTO2620
1710 GOSUB4210
1720 GOSUB3620
1730 Q=Q+1
1740 IFP=148THEN2640
1750 IFP=153THEN1850
1760 IFP=152THEN1820
1770 IFQ=4THEN1800
1780 GOSUB4270
1790 GOTO1720
1800 IFS1+S2=2THEN2640
1810 GOTO2620
1820 S1=S1+1
1830 IFS1=2THEN2640
1840 GOTO1770
1850 S2=S2+1
```

```
1860 IFS2=2THEN2640
1870 GOTO1770
1880 GOSUB4210
1890 GOSUB3620
1900 Q=Q+1
1910 IFP=116THEN2640
1920 IFP=129THEN2020
1930 IFP=130THEN1990
1940 IFQ=4THEN1970
1950 GOSUB4270
1960 GOTO1890
1970 IFS1+S2=2THEN2640
1980 GOTO2620
1990 S1=S1+1
2000 IFS1=2THEN2640
2010 GOTO1940
2020 S2=S2+1
2030 IFS2=2THEN2640
2040 GOTO1940
2050 GOSUB4210
2060 GOSUB3620
2070 Q=Q+1
2080 IFP=153ORP=148THEN2640
2090 IFP=152THEN2130
2100 IFQ=4THEN2620
2110 GOSUB4270
2120 GOTO2060
2130 S1=S1+1
2140 IFS1=2THEN2640
2150 GOTO2100
2160 GOSUB4210
2170 GOSUB3620
2180 Q=Q+1
2190 IFP=130ORP=148GOTO2640
2200 IFP=129THEN2240
2210 IFQ=4THEN2620
2220 GOSUB4270
2230 GOTO2170
2240 S1=S1+1
2250 IF S1=2THEN2640
2260 GOTO2210
2270 P=VPEEK(6143+32*(I-1)+J)
2280 IFP=33ORP=128ORP=136THEN2340
2290 IFT=119THEN2320
2300 IFT=151THEN2330
2310 GOTO2620
2320 IFP=48THEN3050ELSE2620
2330 IFP=38THEN3050ELSE2620
2340 ONCGOTO710,830
2350 IFM=ITHEN2440
2360 IFN<>JTHEN2620
2370 G=(M+I)/2
2380 P=VPEEK(6143+32*(G-1)+N)
2390 IFF=2THEN2410
2400 IFP=33THEN2430
2410 IFP=128ORP=136ORP=40THEN2430
2420 GOTO2620
2430 ONFGOTO850,720
2440 G=(N+J)/2
2450 P=VPEEK(6143+32*(M-1)+G)
2460 GOTO2390
2470 JOG=JOG-1
2480 IFJOG=0THEN440
2490 GOTO510
2500 P=VPEEK(6143+32*(I-1)+J)
2510 IFP=120THEN970
2520 GOTO2620
2530 G=(M+I)/2
2540 P=VPEEK(6143+32*(G-1)+N)
2550 FORS=120TO125
2560 IFP=STHEN970
2570 NEXTS
2580 GOTO2620
2590 G=(N+J)/2
2600 P=VPEEK(6143+32*(M-1)+G)
2610 GOTO2550
2620 PLAY"L2N18V15"
2630 GOTO 510
2640 'INICIO DA MARCHA FUNEBRE
2650 PLAYMF$
2660 FORT=1TO2000:NEXT
2670 IFTI=1THEN2700
2680 GOSUB2740
2690 GOSUB2770
2700 VPOKE(6143+32*(I-1)+J),D
2710 TI=0
2720 IFKUY=1THEN3050
2730 GOTO2470
2740 IFN<10THEN2750ELSE2770
2750 IFN<12THEN2760ELSE2770
2760 D=136
2770 IFM>16THEN2780ELSE2800
2780 IFN>21THEN2790ELSE2800
2790 D=128
2800 RETURN
2810 FORX=1TO3
2820 IFU=53THEN4190
2830 YE=X
2840 IFJ-YE<1THEN2620
2850 P=VPEEK(6143+32*(I-1)+J-YE)
2860 D=120
2870 IFP>111ANDP<116ORP>143ANDP<148THEN2
900
2880 NEXTX
2890 GOTO2620
2900  FORT=1TO13:SOUNDT,23:NEXT:FORS=1TO
500:NEXT:FORS=1TO13:SOUNDS,18:NEXT
2910 FORT=1TO800:NEXT:BEEP
2920 TI=INT(RND(1)*(X+1))+1
2930 J=J-YE
2940 IFTI=1THEN2640
2950 GOTO2470
2960  PLAY"V15L64N41"
2970 RETURN
2980 K$=INKEY$
2990 IF K$=""THEN2980ELSEK=ASC(K$)
3000 IFK=13THENI=INT((Y1+9)/8):J=INT((X1
+9)/8):GOSUB 2960:GOTO3020
3010 IFK>27ANDK<32THENGOSUB4530:GOTO2980
3020 J1=ABS(J-N)
3030 I1=ABS(I-M)
3040 RETURN
3050 V=0
3060 N=0:INTERVALOFF
3070 GOSUB4300
3080 CLS:SCREEN0,,1:COLOR15,1
3090 LOCATE6,10:PRINT"**<QUER JOGAR DE N
OVO S/N>**"
3100 K$=INPUT$(1)
3110 IFK$="S"THEN3130
3120 END
3130 RUN
3140 GOSUB4210
3150 GOSUB3620
3160 Q=Q+1
3170 IF P=153 ORP=152ORP=148THEN3490
3180 IFQ=4THEN2620
3190 GOSUB4270
3200 GOTO3150
3210 GOSUB 4210
3220 GOSUB3620
3230 Q=Q+1
3240 IFP=130ORP=129ORP=116THEN3490
3250 IFQ=4THEN2620
3260 GOSUB4270
3270 GOTO 3220
3280 X=-1
3290 RETURN
3300 X=0
3310 Y=1
3320 RETURN
3330 Y=-1
3340 RETURN
3350 GOSUB4210
3360 GOSUB3620
3370 Q=Q+1
3380 IFP=119THEN2640
3390 IFQ=4THEN2620
3400 GOSUB4270
3410 GOTO3360
3420 GOSUB4210
3430 GOSUB3620
3440 Q=Q+1
3450 IFP=151THEN2640
3460 IFQ=4THEN2620
```

```
3470 GOSUB 4270
3480 GOTO 3430
3490 KUY=1
3500 GOTO2640
3510 IFTER=0THEN2620ELSE1250
3520 IFNAV(R,1)=0THEN3560
3530 IFNAV(R,2)=0THEN3580
3540 IFNAV(R,3)=0THEN3600
3550 RETURN
3560 NAV(R,1)=33
3570 GOTO3530
3580 NAV(R,2)=33
3590 GOTO3540
3600 NAV(R,3)=33
3610 RETURN
3620 P=VPEEK(6143+32*(I-1+X)+J+Y)
3630 RETURN
3640 VPOKE6143+32*(M-1)+N,D
3650 VPOKE6143+32*(I-1)+J,T
3660 GOTO 2470
3670 GOSUB4340:BLOAD TT$,+5000
3680 GOTO100
3690 ME=PEEK(49999!):FORT=50001!TO49998!
+MESTEP3
3700 P1 =PEEK(T)
3710 P2 =PEEK(T+1)
3720 P3 =PEEK(T+2)
3730 VPOKE6143+32*(P1-1)+P2,P3
3740 NEXTT
3750 TE=0
3760 FORT=1TO2:FORS=1TO3:NAV(T,S)=PEEK(5
0500!+TE):TE=TE+1:NEXT:NEXT:JOG=PEEK(505
00!+TE+1):CW=PEEK(50500!+TE+2)
3770
3780 INTERVALON
3790 GOTO470
3800 F=2
3810 GOTO2350
3820 INTERVALOFF:FORT=1TO28
3830 VPOKE6143+T+2,ASC(MID$("INICIANDO.A
GUARDE ALGUNS SEG" ,T,1))
3840 NEXTT:ME=0
3850 FORT=1TO24
3860 FORP=1TO29
3870 G=VPEEK(6143+32*(T-1)+P)
3880 IF G>111ANDG<120ORG>128ANDG<131ORG>
143ANDG<154THEN3920
3890 NEXTP
3900 NEXT
3910 ME$=">>GRAVANDO":GOSUB4340:GOTO3940
3920 ME=ME+3:POKE50000!+ME-2 ,T:POKE5000
0!+ME-1,P:POKE50000!+ME,G
3930 GOTO3890
3940 FORT=1TO2:FORS=1TO3:POKE50500!+TE,N
AV(T,S):TE=TE+1:NEXT:NEXT:POKE50500!+TE+
1,JOG:POKE50500!+TE+2,CW:POKE49999!,ME
3950 BSAVE TT$,49999!,50600!
3960 PRINT:PRINT:PRINT
3970 PRINT" >VOLTAR JOGO- V
          >FIM -F":U$=INPUT$(1)
3980  RESTORE
3990 FORT=1TO13
4000 READA
4010 NEXTT
4020 IFU$="V"OR U$="v"THEN3680
4030 END
4040 GOSUB2980
4050 GOSUB4140
4060 IFDE=1THEN3132
4070 CI=1: CH$="04242424242EFF7E":FORS=0
TO7:VPOKE8*P+S,VAL("&H"+MID$(CH$,CI,2)):
CI=CI+2:NEXT
4080 GOTO2810
4090 GOSUB2980
4100 GOSUB4140
4110 IFDE=1THEN2620
4120 CI=1: CH$="202424242474FF7E":FORS=0
TO7:VPOKE8*P+S,VAL("&H"+MID$(CH$,CI,2)):
CI=CI+2:NEXT
4130 GOTO2810
4140 P=VPEEK(6143+32*(I-1)+J)
4150 DE=0
4800 JU=VPEEK(6143+32*(I-1)+J)
4810 IF JU=33THEN2620
4820 GOTO720
4830 FR$="T130M4500S104L4CL5CFFGGD5L4CD4
L5AFFAFDL4B-L5GEF"
4840 IN$="T140M4500S104L3GGGL2GL4FL3EFFF
L2FL4ED"
4850 MF$="T1102M5500S2L3AL4AL7AL5AO3CO2
L5BBAAA-A"
4860 RETURN
10000 A$="L32GEB"
10010 LOCATE 0,20:LINE INPUT "NOME DO TA
BULEIRO:";TT$:LOCATE0,20:PRINT" ]PREPARE
-SE                    ":PLAYA$
10020 B$=INKEY$:IFB$=""THEN10020
10030 LOCATE0,20:PRINT" ]LENDO
       "
10040 BLOAD TT$,+7000
10050 LOCATE0,20:PRINT" ]AGUARDE UM MOME
NTO":PLAYA$
10060 FORT=0TO2048:B=PEEK(47732!+T):VPOK
ET,B:NEXT
10070 FORT=0TO15:B=PEEK(49781!+T):VPOKE8
196+T,B:NEXT
10080 FORT=0TO731:B=PEEK(47000!+T)
10085 IFU$<>"V"ANDU$<>"v"THEN10200
10090 IFB>111ANDB<116THENB=120
10095 IFB=119ORB=116ORB=129ORB=130THENB=
128
10100 IFB>143ANDB<148THENB=120
10110 IFB=148ORB=151ORB=152ORB=153THENB=
136
10200 VPOKE6178+T,B:NEXT
10300 FORT=0TO27:VPOKE6146+T,0:NEXT
10320 FORT=6146TO6912STEP32:VPOKET,0:VPO
KET+27,0:NEXT
10400 BEEP:RETURN
4380 FORT=1TO8
4390 READ A$
4400 S$=S$+CHR$(VAL("&B"+A$))
4410 NEXTT
4420 SPRITE$(1)=S$
4430 X1=127:Y1=95
4440 INTERVALON: RETURN
4450 DATA 11111000
4460 DATA 10001000
4470 DATA 10001000
4480 DATA 10001000
4490 DATA 11111000
4500 DATA 00000000
4510 DATA 00000000
4520 DATA 00000000
4530 C1=STICK(0):PLAY"V6L64N40"
4540 IFC1=1THENY1=Y1-8
4550 IFC1=2THENY1=Y1-8:X1=X1+8
4560 IFC1=3THENX1=X1+8
4570 IFC1=4THENY1=Y1+8:X1=X1+8
4580 IFC1=5THENY1=Y1+8
4590 IFC1=6THENY1=Y1+8:X1=X1-8
4600 IFC1=7THENX1=X1-8
4610 IFC1=8THENY1=Y1-8:X1=X1-8
4620 IFX1<23THENX1=223
4630 IFX1>223THENX1=23
4640 IFY1<7THENY1=183
4650 IFY1>183THENY1=7
4660 PUT SPRITE0,(X1,Y1),CR,1:RETURN
4670 CR=CR+1:IFCR=16THENCR=0
4680 GOTO 4660
4690 IF CW/2<>INT(CW/2)THENPP=145 ELSE P
P=113
4700 FORT=1TO4:CI=1:CH$="042424242E74FF7
E":FORS=0TO7:VPOKE8*PP+S,VAL("&H"+MID$(C
H$,CI,2)):CI=CI+2:NEXT
4710 CI=1:CH$="202424242474FF7E":FORS=0T
07:VPOKE8*PP+S,VAL("&H"+MID$(CH$,CI,2)):
CI=CI+2:NEXT:NEXT
4720 GOTO510
4730 CLS:FORT=0TO38:LOCATET,1,0:PRINTCHR
$(219);:LOCATET,22:PRINTCHR$(219);:NEXT
4740 FORT=1TO21:LOCATE0,T:PRINTCHR$(219)
;:LOCATE 40,T:PRINTCHR$(219);:NEXT
4750 LOCATE 1,2:PRINT"     C.P.U. APRESE
4160 IFP>111ANDP<116ORP>143ANDP<148THEN4
180
4170 DE=1
4180 RETURN
4190 YE=-X
4200 GOTO2840
4210 Q=0
4220 S2=0
4230 S1=0
4240 X=1
4250 Y=0
4260 RETURN
4270 ONQGOSUB3280,3300,3330
4280 RETURN
4290 ON K-47 GOTO3820,1040,1390,1600,404
0,4090,4690
4300 'ROTINA DE SONS DE VITORIA
4310 IF CW/2<>INT(CW/2)THENPLAYFR$
  ELSE PLAYIN$
4320 K$=INKEY$:IFK$=""THEN4320
4330 RETURN
4340 SCREEN0:COLOR1,15:LOCATE0,10:LINE I
NPUT "DE O NOME:";TT$:LOCATE0,10:PRINT">
PREPARE-SE E PRESSIONE RETURN";
4350 BV$=INPUT$(1)
4360 IFASC(BV$)<>13THEN4350ELSE  PRINT:P
RINT:PRINTME$:RETURN
4370 RESTORE 4450
NTA: TRAFALGAR PLUS":LOCATE2,10:PRINT"PO
R: FERNANDO LEIBEL":FORE=1TO1000:NEXT:LO
CATE 2,12:PRINT"musica:F.LIPORACE"
4760 LOCATE2 ,18:PRINT"JOGO NOVO (N)":LO
CATE2,19:PRINT"JOGO VELHO (V)"
4770 FORT=2TO20:LOCATE24,T:PRINT"TRAFALG
AR PLUS":U$=INKEY$:IFU$<>""THEN4790
4780  BEEP:LOCATE24,T  :PRINTSPC(14) :NE
XT:GOTO 4770
4790 GOTO80
```

# A INFORMÁTICA NA MEDICINA

*FERNANDO LEIBEL*

Desde o início da era dos computadores, os seres humanos começaram a imaginar as suas aplicações na Medicina, principalmente os leigos que tinham verdadeiros pesadelos com computadores encostando seus braços eletrônicos em suas costas e pedindo-lhes que falassem "trinta e três".

Você, leitor, não se assute. A cena que acabei de descrever ainda faz parte dos livros de ficção e, possivelmente, nem eu nem você viveremos o bastante para vê-la acontecer.

**A realidade atual**

Saindo do mundo de conto de fadas e passando para o mundo real e palpável do século XX, vemos que os computadores realizam tarefas bem mais simples, porém não menos importantes.

Na medicina o computador pode ser utilizado de várias formas, como veremos abaixo:

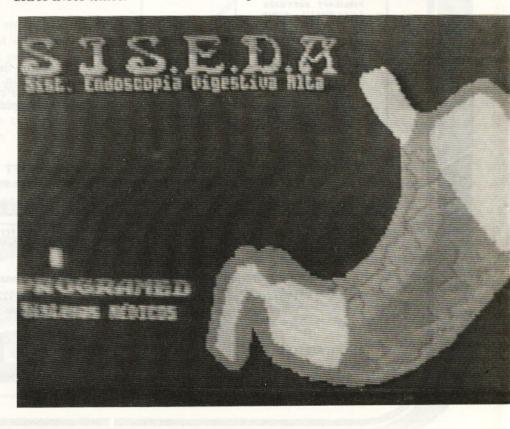
– Como secretárias, que tomam conta da agenda de pacientes e da contabilidade dos consultórios, além de poder dar uma mãozinha no imposto de renda.

– No auxílio aos laudos, imprimindo, arquivando e permitindo pesquisas das mais diversas formas.

– No ensino aos médicos ainda em formação, através de simulações dos sistemas fisiológicos, reações orgânicas às drogas ou como verdadeiros livros textos.

– No controle de estoque dos grandes hospitais e milhares de outras aplicações.

Um exemplo prático de sistema de auxílio é o SIS.E.D.A., que vem a ser um sistema dedicado à Endoscopia Digestiva Alta, idealizado e desenvolvido por mim. O programa é capaz de arquivar os laudos, emitir uma lista de laudos por nome, data, diagnóstico histopatológico, além de possuir fichas completas para descrição endoscópica, entre outras coisas.

Sem dúvida nenhuma, o ponto alto deste programa é a facilidade de operação, tendo sido apresentado ao público no último Congresso Brasileiro de Gastroenterologia e Endoscopia Digestiva Alta. A facilidade de utilização foi comprovada, pois médicos que nunca haviam operado com o sistema, e muito menos com computadores, efetuaram consultas sem nenhuma dificuldade após poucos minutos. O SIS.E.D.A. está disponível nas versões MSX e IBM PC.

### Os médicos e os computadores

Os médicos possuem uma certa resistência aos computadores. Resistência esta normal, pois, para eles, a informática é um mundo desconhecido, cheio de perigos e armadilhas à sua espera. De certa forma, não deixam de ter razão, pois a escolha errada de computadores, de programas ou de ambos pode significar grandes problemas, ao invés de soluções.

Os maiores culpados desse temor são os próprios profissionais de informática, que não fazem a mínima questão de simplificar os procedimentos, preferindo, ao contrário, jogar uma aura de genialidade em cima dos produtos, que, na verdade, são é mal acabados e, por isso, de difícil utilização.

### As vantagens de se ter um computador

A melhor forma de mostrar as vantagens de se ter um computador é, sem dúvida nenhuma, através de exemplos que fazem parte do cotidiano médico.

O Dr. X é endoscopista há dez anos. Seus laudos são datilografados por sua fiel secretária em três vias, sendo uma para o INPS, outra para o paciente e a última para ele guardar em seu arquivo, do qual tem muito orgulho e que já possui dez mil laudos arquivados, ocupando uma área de dois metros quadrados.

Um dia o Dr. X resolveu fazer uma pesquisa de quantos pacientes possuem esofagite grau 1 associado a bulbite superficial. Após três meses de pesquisa, o Dr. X desiste e chega à conclusão que seu arquivo só serve para ocupar espaço.

O Dr. Y é endoscopista também há dez anos, sendo que, há dois anos, adquiriu um computador e um sistema para endoscopia digestiva alta. Este sistema também libera laudos em três vias, mantendo um armazenado em disquete.

O Dr. Y, logo que adquiriu o sistema, passou todo o seu antigo e enorme arquivo para disquetes, sendo que, como o Dr. X, o Dr. Y também possui dez mil laudos, porém o seu arquivo agora ocupa dez disquetes ou dez centímetros quadrados.

Em um certo dia, o Dr. Y resolveu pesquisar quantos pacientes possuíam esofagite grau 2 associado a gastrite. Após trinta minutos, o Dr. Y chegou à conclusão que 12% dos seus pacientes possuíam esta patologia.

Acredito que os dois exemplos acima servem para demonstrar a vantagem de se ter um computador.

### Os computadores nos aparelhos médicos

Os computadores praticamente invadiram a maioria dos aparelhos que os médicos utilizam diariamente.

Hoje, eles estão presentes desde em um simples monitor cardíaco até sofisticados aparelhos como o tomógrafo computarizado. Todos os aparelhos, por exemplo, ultra e ecocardiográficos possuem no seu interior verdadeiros computadores com softs sofisticadíssimos que tornam sua utilização simples e eficiente. Nos mais avançados centros de tratamento intensivo,o computador é capaz de monitorar o paciente 24 horas por dia e avisar ao médico qualquer alteração nos seus sinais vitais. O computador é hoje peça fundamental na luta da vida contra a morte.

### O futuro da informática na medicina

O limite dos computadores é, sem dúvida, a imaginação dos seres humanos. Há tempo atrás, lendo uma reportagem sobre os biochips, chips vivos, minha imaginação deu um vôo e fiquei pensando se, com estes novos chips, não seria possível a reconstrução de conexões nervosas perdidas, por exemplo, num acidente automobilístico. Com isto, poderíamos impedir que o indivíduo ficasse paraplégico.

Realmente, o parágrafo acima não passa de um sonho, mas a viagem à Lua também foi um sonho para Júlio Verne. ■

*Fernando Leibel é médico e autor de diversos softwares entre eles o Enigma dos Deuses, Trafalgar, Polaris II, Missão Estrelar e SIS.E.D.A. Atualmente é o responsável pela informatização do CTI da Santa Casa de Misericórdia do Rio de Janeiro.*

# CPU LIVROS

■

ANÁLISE DO LIVRO

**Como usar o SQL do dBASE IV**
**Datalógica**
**Editora McGraw-Hill**

O livro COMO USAR A SQL DO dBASE IV, editado pela McGraw-Hill, destina-se a usuários do dBASE IV, ou que estejam migrando para esta linguagem, e aborda o uso do SQL nesta linguagem, já que faz parte de suas opções. SQL significa: Structured Query Language, de difícil tradução para o português. Algo como: Linguagem Estruturada Interrogativa.

O SQL foi criado pela IBM, a partir das pesquisas de E. F. Codd, que introduziu o conceito de banco de dados relacionais a partir de um ramo da matemática, a álgebra relacional, que, trocada em miúdos, trata de dados organizados em tabelas.

Os microcomputadores passaram a dispor do SQL gradativamente, a partir do ORACLE, depois OPEN ACCESS, PARADOX e, agora, dBASE IV. O SQL usado pelo dBASE IV é o que mais se aproxima do padrão adotado pela IBM, o que se configura em um idioma comum aos ambientes micro e mainframes. Mas, segundo o próprio autor, não se deve esperar grandes resultados desta interface de comunicação, já que o SQL ainda está longe de ser um padrão acabado.

Para exemplificar a utilização do SQL junto ao dBASE IV, o autor desenvolve um pequeno sistema para controle de um hotel. Não chega a ser um sistema acabado, pois usa apenas os dados e operações indispensáveis ao entendimento dos comandos apresentados, mas cumpre a sua função de mostrar o uso da linguagem associada aos recursos do dBASE.

**CONCLUSÃO:**

**O livro cumpre seus objetivos de explicar tanto SQL quanto seu uso junto ao dBASE, porém o efeito que me causou, após a leitura, não foi o que, provavelmente, seria esperado pelo autor, pois são tantas as restrições e tão poucas as vantagens, que, dificilmente, optaria por usar tal linguagem. Naturalmente, esta não seria a opinião de programadores mais voltados para a portabilidade de grandes sistemas, onde o SQL pode vir a ser indispensável.**

■

ANÁLISE DO LIVRO

**80386 – Guia Técnico do Programador**
**Chris H. Pappas – William N. Murray, III**
**Editora McGraw-Hill**

O 80386 é um microprocessador de 32 bits, projetado para trabalhar com sistemas operacionais otimizados para tarefas múltiplas. Pode endereçar até 4 gigabytes de memória física e 64 terabytes de memória virtual.

A definição acima foi retirada do livro 80386 – GUIA TÉCNICO DO PROGRAMADOR, auto-definido como um manual técnico do 80386 para programadores, principalmente os que utilizam a linguagem "C" e o assembler, das quais há inúmeros exemplos nos diversos capítulos.

É quase que um manual obrigatório para o programador, caso queira se aventurar a trabalhar a nível de microprocessador. Apesar do pressuposto de que seu leitor deve ter um bom conhecimento das famílias de processadores 8086 e 80286, além de bom domínio das linguagens citadas, o livro evita termos excessivamente técnicos e seus exemplos práticos se adaptam, em grande parte, a aplicações banais, tais como paginação de memória, impressão de gráficos e cálculos matemáticos, entre outros.

Os mnemônicos do assembler 80386 são comentados e explicados de forma prática e de fácil assimilação, sendo que as expressões originais, em inglês, são sempre mantidas entre parênteses, mesmo quando o termo é traduzido.

Todos os exemplos são comentados em cada linha de instrução, trazendo referências sobre os compiladores e linkeditores usados no desenvolvimento dos mesmos.

Além do 80386, o manual se expande aos "chips" de suporte que complementam este microprocessador, ou seja, o co-processador numérico 80387, o controlador de "caché" 82385 e o ADM 82380, um controlador de periféricos. A cada um foi dedicado um capítulo inteiro, além das ocasiões em que seu uso nos exemplos exigiu explicações sobre os mesmos.

**CONCLUSÃO:**

**É um livro para iniciados. Não é leitura fácil para qualquer programador, mas pode servir tanto para o profissional quanto ao programador em desenvolvimento que queira se aprofundar mais nos segredos e uso direto da linguagem de máquina. Não é um manual definitivo, uma vez que cita freqüentemente outras obras para complementar assuntos nos quais não se aprofunda, mas, de qualquer forma, irá freqüentar a prateleira da maioria dos usuários que programam em ambiente 386.**

■

ANÁLISE DO LIVRO

**dBASE IV**
**Primeiro livro de iniciação e interação ao dBase IV**
**Datalógica**
**Editora McGraw-Hill**

Antigamente, a informática era tratada como um grande e desconhecido universo, reservado apenas a alguns eleitos que operavam máquinas fantásticas. Os usuários, praticamente, não existiam, ou eram meros espectadores.

A tendência, cada vez mais forte, dos programas e dos próprios computadores, é retirar esta aura de mistério e dificuldades, permitindo que a informática possa ser utilizada por todos, sem necessidade de cursos complicados e, principalmente, sem ter que adaptar o homem à máquina: ela é que tem que se adaptar a nós.

O sistema de gerenciamento de dados dBASE tornou-se uma das

linguagens mais conhecidas e utilizadas em qualquer assunto que envolva manipulação de dados. No entanto, sempre sofreu da síndrome do desconhecido, que aterroriza qualquer candidato a usuário do programa.

Já no dBASE III Plus, o utilitário ASSIST facilitava bastante a interação máquina/usuário e o dBASE IV foi lançado com a intenção de diminuir ainda mais esta distância. Se vai conseguir, só o tempo dirá, mas o nosso assunto principal é o livro "dBASE IV FOR FIRST TIME USER'S", da McGraw-Hill, que se apresenta como o manual dos "marinheiros de primeira viagem" para quem pretenda se iniciar no dBASE.

Assim como a informática, o livro assusta pelo tamanho: 572 páginas de texto e figuras explicativas tiradas do próprio programa e, o que é melhor, da versão em português do dBASE IV. Não se deixe impressionar. O tamanho do livro é em função da quantidade de opções que o programa oferece. Você não terá que ler o livro todo, pelo menos não de uma vez.

Desde o início, a preocupação é em explicar detalhadamente e com exemplos práticos cada passo da utilização dos recursos do programa, o que irá dar, gradativamente, ao usuário o conhecimento e domínio das facilidades que são oferecidas.

O sumário do livro facilitará a procura de situações específicas para os que já têm intimidade com o dBASE, mas não conhecem o dBASE IV. Acompanham, também, alguns apêndices que poderão familiarizar os novos usuários que não conhecem nem mesmo o sistema DOS e alguns periféricos exigidos pelo dBASE IV, tais como o disco rígido, ou "winchester" como é mais conhecido.

**CONCLUSÃO:**

**É um excelente manual de iniciação. Contém exemplos de fácil compreensão e um texto agradável de se acompanhar. Não seria de muita valia aos profissionais já tarimbados, uma vez que existem outros livros para estes, mas é um livro indispensável para quem deseja se iniciar em um programa que se propõe a diminuir, até o limite do possível, o trabalho de programação.**

■

ANÁLISE DO LIVRO

**Turbo Pascal avançado**
**Guia do usuário**
**Herbert Schildt**
**Editora McGraw-Hill**

O PASCAL foi criado originalmente para ser uma linguagem educacional, que proporcionaria aos estudantes de informática o desenvolvimento de bons hábitos na programação pelo uso de uma linguagem estruturada. Com o passar do tempo, devido às facilidades que oferecia, o PASCAL foi sendo cada vez mais utilizado na elaboração de programas comerciais e científicos, fazendo surgir o TURBO PASCAL.

O livro TURBO PASCAL AVANÇADO, da editora McGraw-Hill, vem auxiliar os programadores já íntimos com a linguagem a desenvolver e utilizar conceitos mais avançados do TURBO PASCAL. Não seria recomendado para usuários iniciantes, a não ser para uso futuro, pois utiliza exemplos e conceitos que exigem um certo grau de conhecimento da linguagem.

Usando como base o TURBO PASCAL da BORLAND, que traz em seus capítulos exemplos e soluções para diversas tarefas de áreas diferentes, tais como ordenação e comunicação com o sistema operacional. Além disso, aborda assuntos como as extensões Turbo Database Toolbox e Graphix Toolbox, além de conversões de BASIC e "C" para TURBO PASCAL.

As diferenças entre as versões 3.0, 4.0, e também a nova versão 5.0 do Turbo da Borland, são abordadas em forma de guias de consulta, fornecendo ao usuário uma visão geral das opções disponíveis em cada versão.

**CONCLUSÃO:**

**Um excelente manual e guia de referência para profissionais experientes, que possibilitará tirar melhor proveito de seu TURBO. Possui 405 páginas de informações úteis e ótimo nível técnico.**

■

ANÁLISE DO LIVRO

**Cobol para Microcomputadores**
**Mutsuo Ono**
**Editora McGraw-Hill**

A linguagem COBOL (Common Business Oriented Language ou Linguagem de Computador Orientada para Negócios), apesar de defasada em relação a novas linguagens mais potentes, com as quais se pode atingir melhores resultados, resiste bravamente como linguagem base para a maioria das aplicações comerciais. Talvez mais pela quantidade de sistemas em COBOL já existentes, que pelas suas qualidades.

O livro COBOL para microcomputadores, da editora McGraw-Hill, e de autoria de Mutsuo Ono, propõe-se a explicar o COBOL para iniciantes e estudantes de informática, e o faz de forma concreta.

Tratando basicamente dos comandos e estrutura dos programas em COBOL, o livro faz as vezes de um manual, só que com uma abordagem mais didática.

A obra traz muitos exemplos práticos, que servem de suporte aos conceitos apresentados, além de explicação dos principais comandos e funções do COBOL 80/MS/MB.

As definições de instruções, assim como estruturas dos tipos de arquivo usados pelo COBOL, não chegam a ser detalhadas. Normalmente, são definidos em rápidas palavras, ao contrário dos programas, que são melhor comentados e explicados.

**CONCLUSÃO:**

**É um livro basicamente voltado para o ensino do COBOL. Não teria outra função, já que não trata de nenhum tipo particular da linguagem e nem se aprofunda o bastante para se configurar um manual de consulta. Tem seu lugar garantido nas prateleiras das bibliotecas das universidades e cursos de informática.** ■

## ROBOCOP & RAMBO III

A NEMESIS traz com a exclusividade de costume, as ultimas novidades em MOVIE GAMES para a linha MSX-1 e MSX-2:

### ROBOCOP

Baseado no sensacional filme, este jogo reproduz com fidelidade as principais cenas vividas pelo nosso heroi !!!!

### RAMBO III

Como no filme campeao de bilheteria, sua missao e´ resgatar o coronel Trautman.

CADA UM POR APENAS CZ$ 9,00

### THE STRIKE FORCE HARRIER

VERSAO DUPLA MSX1 E MSX2 64KB

Um super simulador de combate aereo com o mais famoso caca-bombardeiro vertical ingles. Apenas em disco por Cz$ 8,00.

### SCORE 2020

Imagine um PIMBALL com diversos cenarios e obstaculos dos mais incriveis. Prepare-se!! Apenas em disco por Cz$ 7,00.

### LEGION XEVIOUS

Se voce esta´ a fim de um jogo espacial como antigamente, esta e a sua grande chance! Apenas em disco por Cz$ 8,00.

## AS NOVIDADES PARA SEU MSX

### HERCULES THE SLAYER OF DAMAGE

Uma sensacional aventura mitologica onde sua forca e´a sua principal arma de combate. Apenas em disco por Cz$ 7,00.

### STRIP POKER II PLUS

A melhor versao de strippoker existente para a linha MSX. Apenas em disco por Cz$ 7,00.

### BARBARIAN

Um jogo altamente inovador no estilo "ROLE PLAYING GAME". Apenas em disco por Cz$ 8,00.

### THE COMMANDO TRACER

Uma super aventura espacial com um veiculo incomum e suas diversas armas e recursos. Apenas em disco por Cz$ 8,00.

### THOR PRINCE OF OLIMPUS

O super-heroi da MARVEL e sua primeira aventura para os computadores MSX. Nao perca !!! Apenas em disco por Cz$ 7,00.

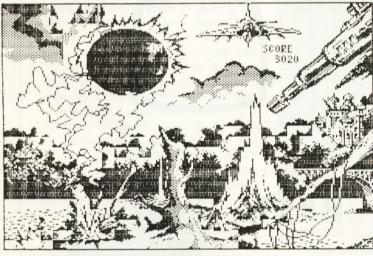
### THE F-15 STRIKE EAGLE

Um dos melhores simuladores de voo de todos os tempos, finalmente adaptado para o MSX. Muita acao espera por voce !!! Nao perca este lancamento !!! Apenas em disco por Cz$ 7,00.

## SPECIAL GAME PACK 12

BARBARIAN, SCORE 3020, F-15 STRIKE EAGLE e THE COMMANDO TRACER. Um super pacote !!! Apenas em disco por Cz$ 18,00

## NOVIDADES SUPER ESPECIAIS

KNIGHTMARE III..MSX1-MEGARAM;
FAMILY BILIARDS.MSX2-MEGARAM;
OUTRUN 256 KB...MSX2-MEGARAM;
TOPPLE ZIP II...MSX2-MEGARAM;
STRIKE F. HARRIER..MSX2-64kb;
CROSS BLAIN.......MSX2-128kb;
FANTAZY ZONE......MSX2-128kb;
DEEP FOREST.......MSX2-128kb.
CADA UM EM DISCO 50'Cz$ 8,00.

## CAPAS PARA COMPUTADORES

Proteja o seu equipamento da poeira e da umidade:

EXPERT (CPU+TECLADO).Cz$ 18,00
HOTBIT ou TALENT....Cz$ 12,00
CAPA PARA DRIVE.....Cz$ 8,00
CAPA PARA LADY/MTA..Cz$ 12,00

## NEMESIS INFORMATICA LTDA.

Envie VALE POSTAL ou CHEQUE NOMINAL a NEMESIS INFORMATICA Caixa postal 4.583 Cep 20.001 Rio de Janeiro - RJ ou venha pessoalmente na: Rua Sete de Setembro 92/1910 CENTRO - RJ.

## SPECIAL GAME PACK Nº 10

Um pacote super especial com as seguintes novidades:

HERCULES THE SLAYER OF DAMAGE STRIP POKER II PLUS, GHOST OF DANDELION e BOB 007.
Apenas em disco por Cz$ 15,00

## SPECIAL GAME PACK Nº 11

THE STRIKE FORCE HARRIER, THE LEGION XEVIOUS, THOR PRINCE OF OLYMNPUS e CRAZY TRANSIT.
Apenas em disco por Cz$ 15,00

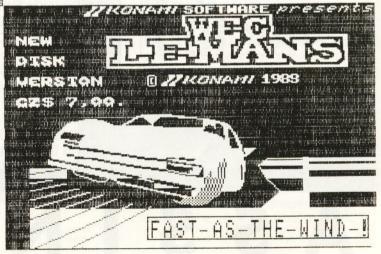
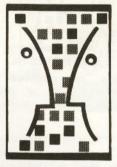
AGUARDE OS LANCAMENTOS:
DOUBLE DRAGON E AFTERBURNER

## NOVIDADES P/ MSX PAGE MAKER

MSX PAGE MAKER FONTS 3 e PAGE MAKER FONTS 4. Mais uma variedade inedita de:

LETRAS, LETRAS, LETRAS, ETC.

MSX PAGE MAKER CARTOONS 2 e MSX PAGE MAKER TITLES nº1 com uma infinidade de figuras para compor suas artes:

CADA UM EM DISCO - Cz$ 12,40.

ENTREVISTA

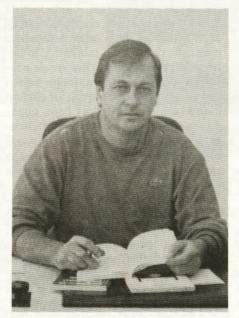
## Pierluigi Piazzi, 46, químico industrial, físico, professor e o Diretor-Editorial da Editora Aleph e também um dos maiores motivadores do MSX no Brasil.

**Antes de se dedicar ao MSX, Pierluigi também contribuiu para o desenvolvimento de outras linhas de computadores (TK e Apple), tendo sido o Diretor-Editor da revista Microhobby, dirigida aos usuários da linha TK.**

*** Reserva de Mercado e Lei de Informática. Vantagens ou prejuízos para o Brasil.**

A Lei de Informática tem efeito análogo ao da Lei-Seca que foi instituída nos EEUU entre as duas grandes guerras. Muito bem intencionada, ao tentar resolver um problema, criou outros piores. A Lei-Seca quis acabar com o alcoolismo e, na realidade, acabou criando uma tamanha rede de contrabandistas, produtores clandestinos, gangsters, policiais e políticos corruptos, que teve que ser revogada. A Lei de Informática e a Reserva de Mercado foram criadas para proteger e desenvolver a indústria nacional de microinformática (intenções altamente louváveis).

No fim, acabaram gerando um quase irrecuperável atraso tecnológico, proteção cartorial de empresários incompetentes, corrupção, contrabando e pirataria. Hoje, o lobby que pressiona os congressistas em Brasília para manter a Reserva de Mercado é financiado não por industriais, mas por agentes alfandegários corruptos, que permitem uma indústria do contrabando que hoje é o maior fornecedor do mercado de microinformática.

E que não venham me acusar de entreguista e anti-nacionalista, logo eu, que, por questões de idealismo, nunca publiquei sequer um livro traduzido.

SUA EDITORA POSSUI DIVERSOS LIVROS PUBLICADOS PARA COMPUTADORES DA LINHA MSX, TODOS OS AUTORES NACIONAIS. ALÉM DA PUBLICAÇÃO DE LIVROS, A ALEPH TAMBÉM PUBLICA DIVERSOS MANUAIS DE EQUIPAMENTOS, COMO O QUE ACOMPANHA O EXPERT E O DRIVE DA SHARP, ENTRE OUTROS. NA ÁREA DE LIVROS, SEUS MAIORES SUCESSOS SÃO O "APROFUNDANDO-SE NO MSX" E "100 DICAS PARA MSX".

Sob a bandeira de um nacionalismo hipócrita se esconde um esquema de corrupção e podridão que faria inveja a qualquer Máfia.

*** MSX. O que estas três letras, famosas lá fora, representam para o mercado de Informática brasileiro?**

Na realidade, existem duas informáticas no Brasil: uma é a informática oficial, das empresas e dos ricos, abarrotada de pomposos e engravatados "analistas", "assessores", "consultores", etc. Para estes, um IBM PC XT é o mínimo de hardware decente para se começar a pensar em algo sério. Do outro lado, temos informática alternativa, a dos usuários de poder aquisitivo nível-Brasil. Com a quantia com que um usuário americano compra um XT para pôr em casa, o brasileiro compra a casa!

Para estes usuários, o MSX representa o micro de melhor relação custo/benefício: é barato, acessível a uma faixa grande do nosso mercado e tem uma arquitetura extremamente inteligente e versátil. Enquanto não fabricarem PCs a um quinto do preço atual, o MSX continuará sendo a melhor opção para usuários domésticos, pequenos empresários e profissionais liberais.

Até lá, os próprios "engravatados" da outra informática acabarão aceitando o MSX, coisa que não fizeram até agora por preconceito, ignorância, pura burrice (tem analista que acha que um micro de 16 bits tem o "dobro" da capacidade de um de 8!), e por pressão dos fabricantes de "clones-PC", que morrem de medo de MSX. Hoje se vende muito PC

a usuários desavisados, que poderiam muito bem suprir suas necessidades com um MSX a um custo absurdamente menor.

*** O TK e o Apple já tiveram o seu tempo de glória. E o MSX?**

O Apple ainda tem seus fãs, mas já está obsoleto, se comparado ao MSX (que, aliás, foi projetado usando o Apple como contra-exemplo).

O TK só morreu por falta de visão da Microdigital e da Prológica. Na minha opinião, o Sinclair é uma máquina maravilhosa e hoje poderíamos ter mais de um milhão delas nas escolas e nos lares, iniciando crianças para o maravilhoso mundo da informática.

O MSX começa, agora, a viver seu momento de glória; o melhor ainda está por vir.

*** Em todos os seus artigos e editoriais o Sr. é um ferrenho defensor do MSX. Por que optou por um MSX e não um PC, por exemplo? Seria apenas uma questão de custo dos equipamentos?**

A minha formação (e vocação) é de professor: gosto de trabalhar com pessoas que têm vontade de aprender. Trabalhando com um público de menor poder aquisitivo, defronto-me com usuários que tentam suprir a falta de verbas com criatividade, engenhosidade e inteligência.

Quantas pessoas sabem (ou querem aprender) programar um TK ou um MSX? Inúmeras!

Agora, eu pergunto: quantas pessoas têm interesse em aprender a programar um PC? A maioria dos usuários de PC são "usadores de software", ou seja, suprem sua falta de criatividade comprando a inteligência alheia.

*** Calcula-se, hoje, em aproximadamente 140.000 MSXs vendidos no Brasil. Existe algum fator que possibilite avaliar qual o percentual dessas máquinas que estão sendo usadas? Dos equipamentos atualmente em uso, qual o percentual que está equipado com, pelo menos, uma unidade de disco?**

Somando todos os livros que minha editora já vendeu, e partindo do pressuposto de que quem compra um livro sobre MSX é porque quer usá-lo, estimo em 80% a quantidade de máquinas "não engavetadas". Destas, aproximadamente 1/5 tem unidade de disco.

*** Um dos seus artigos publicados em CPU afirmou que os periféricos para MSX não são caros; o que é barato é o computador em si. Em termos práticos, o que isto vem significando para o usuário que comprou um computador cujos periféricos, muitas vezes, possuem um custo superior ao de uma CPU nova.**

Paradoxalmente, a culpa dos periféricos serem caros é um pouco do usuário. Vou exemplificar: no começo do MSX, nem a Sharp nem a Gradiente produziram unidades de disco. Uma pequena empresa cearense, a Microsol, num lance arrojado e pioneiro, começou a fabricar este periférico para o MSX. Diga-se de passagem, aqui na editora, tenho vários drives da Microsol que funcionam muito bem há anos! Pois bem, quando um usuário ligava para a Gradiente pedindo uma unidade de disco e este indicava a Microsol, ouvia a resposta "Não, não, eu quero a da Gradiente!"

Deixando de lado a discussão sobre confiança na marca, veja o que isso gerou: a Microsol não cresceu, continuou produzindo em moldes quase artesanais, tendo que comprar o acionador de terceiros e não conseguiu baratear seus custos. Quem saiu prejudicado? O próprio usuário!

As coisas só são baratas se produzidas em grande escala. Se eu conseguisse vender, de cada um dos meus títulos, o dobro do que vendo, poderia vendê-los a um preço 30 a 40% menor!

O resto da culpa cabe à famosa Lei de Informática, que já discutimos: se o fabricante pudesse importar componentes mais livremente, poderia praticar preços mais decentes.

Aliás, hoje, estão todos entre a cruz e a caldeira: os que são suficientemente pequenos para poder usar componentes contrabandeados, não podem usufruir de economia de escala, o barato das grandes quantidades; os que são grandes, usufruindo da economia de escala, são muito fiscalizados (e patrulhados!) e não podem usar o contrabando, tendo que desenvolver tecnologia própria, que custa os olhos da cara, e acaba reencarecendo o produto!

*** A Gradiente prepara-se para lançar no mercado um MSX voltado para o uso profissional, que será comercializado com um drive de 3 1/2" embutido. Para completar, também será lançado um cartão de 80 colunas e processador de texto, além de um modem com discagem automática. Depois de um período de silêncio tão grande, com que olhos podemos ver estes novos lançamentos, prometidos há tanto tempo?**

Claro! Antes tarde do que nunca! Aliás, deixa aproveitar para dar uma de futurólogo: o DDPLUS (MSX com drive embutido) vai fazer sucesso não só no mercado profissional! No próprio mercado de lazer, ele vai superar o MSX pelado! Quem tem uma idéia do que aconteceu com o Amstrad na Europa, sabe do que estou falando!

*** No início da comercialização do MSX no Brasil, o mercado de software para esta linha era composto quase que exclusivamente por piratas juniores, que formavam as famosas Softroom, ou seja, aquele garoto que, para complementar a sua mesada, resolvia vender programas. Como vai ficar este mercado?**

O mercado de software vai ficar dividido em dois: os garotos da softhome vão continuar pirateando joguinhos da Konami e algumas empresas mais sérias vão produzir utilitários e aplicativos (aliás, já estão!).

Aproveitando a oportunidade, é conveniente avisar esses garotos que, se eles tentarem invadir o outro mercado com esquemas "bucaneiros", vão levar umas palmadas bem ardidas! O mercado de softhouse pode e deve ser moralizado para que o produto nacional tenha um retorno do seu investimento e sinta-se incentivado a produzir mais.

*** O MSX 2 já é uma realidade lá fora e suas vantagens, dependendo da utilização que lhe for dada, são**

**inúmeras sobre o MSX 1. É possível arricar um palpite de quando teremos no Brasil a nova versão do MSX, sem ser o transformado.**

Veja bem, as vantagens do MSX 2 sobre o MSX 1 não são tantas assim! Ele é igual ao 1, tendo apenas mais (e maravilhosos) recursos gráficos.

Para profissionais da área de comunicações (publicidade, arte, vídeo, etc.) ele é insuperável, mas, para outras áreas, ele não é tão superior assim.

Mesmo assim, se o mercado reagir bem ao "renascimento" do MSX em 89, provavelmente o teremos em 90.

Mas tudo pode acontecer: há boatos de um terceiro fabricante que substituiria a Sharp na concorrência à Gradiente.

Pode ser que ele resolva entrar na briga para valer. Neste caso, talvez comece já com o MSX 2, de cara!

*** Transformar MSX 1 em MSX 2 vale a pena?**

Para os profissionais das áreas que já citei, sem dúvida! Para os outros usuários, pode valer pela novidade e pelo gostinho de se sentir "na vanguarda".

*** Como diretor da Aleph, fica-lhe fácil ver o que o usuário de MSX está querendo, pois certos livros passam a vender mais do que outros. Atualmente, que tipo de informação o usuário mais procura? Tem sido notada alguma mudança desde o tempo que o MSX foi lançado até os dias de hoje?**

O que posso se tir não é bem o que o usuário está querendo. Posso sentir, isso sim, o que o "usuário que lê livros" quer!

Isso me coloca na frente de uma fatia diferenciada da massa total de usuários.

Os meus leitores são pessoas inteligentes, curiosos e que têm muita vontade de aprender. Desde seu lançamento, o livro mais procurado, até hoje, é o "Aprofundando-se no MSX", o que caracteriza um usuário hobbista, criativo. Para atender este usuário, estamos terminando um "Aprofundando-se nos Periféricos do MSX" que, tenho certeza, vai re-editar o sucesso do seu antecessor. Em contra-partida, quando lançamos o "100 Dicas para MSX", tentamos concentrar a maior quantidade possível de informações no espaço disponível. Apesar do enorme sucesso que ele teve, uma crítica foi constante: "informava" muito mas "formava" pouco.

Por isso, quando lançamos o "+50 Dicas para MSX", reduzimos o número de informações, mas aumentamos em muito as explicações. Dito e feito: todo mundo satisfeito!

Resumindo, o nosso leitor quer aprender, aprender, aprender!

E isso, para um professor, é extremamente gratificante! ■

**ASSEMBLER BASIC**

# CURSOR PISCANTE NO MSX

*LUIZ CARLOS BITTENCOURT*

Já dizia o filósofo popular: – "Quem não se comunica..." O Interpretador BASIC é um programa originalmente incorporado ao MSX, com a finalidade de possibilitar ao usuário uma comunicação com o micro através de uma linguagem ao mesmo tempo simples e poderosa – o BASIC/MSX.

O Z-80 passa a executar o programa "Interpretador BASIC" logo após a máquina ter sido ligada, e este fica, então, "aguardando" as ordens do usuário.

Para capturar as mensagens introduzidas via TECLADO, o Interpretador BASIC fica permanentemente consultando uma "FILA DE ENTRADA", onde são colocados os caracteres correspondentes às teclas acionadas pelo usuário (o BUFFER do Teclado).

A colocação dos caracteres nesta fila é feita por um PROCEDIMENTO INDEPENDENTE DO PROGRAMA QUE ESTÁ SENDO EXECUTADO pelo Z-80.

Isto é feito da seguinte maneira: em intervalos regulares de tempo (60 vezes a cada segundo), o Z-80 "interrompe" a tarefa que está executando e "consulta" o teclado para verificar se alguma tecla está sendo pressionada pelo usuário (com a ajuda de outro processador, o PPI), colocando no BUFFER o código correspondente à tecla acionada em caso afirmativo. Em seguida, o Z-80 retorna à tarefa que estava sendo executada, no ponto em que foi interrompida.

Com esta técnica, o MSX continua aceitando a introdução de caracteres, mesmo paralelamente à execução de outra tarefa pelo Z-80.

```
Programa CURSOR PISCANTE em ASSEMBLER Z-80
(A Direita a representação hexadecimal das
instruções em linguagem de máquina Z-80)

 10 ; Rotina para CURSOR PISCANTE / MSX
 20            ORG  #FA75
 30            LD   HL,#09DA        21DA09
 40            EX   (SP),HL         E3
 50            LD   IX,#0A27        DD21270A
 60            LD   IY,#CARATER     FD2181FA
 70 CARATER:   CALL #0D6A           CD6A0D
 80            JR   Z,PISCA         2809
 90            POP  AF              F1
100            CP   #09             FE09
110            JP   Z,#10E1         CAE110
120            JP   #10DE           C3DE10
130 PISCA:     JR   CARATER         18F0
140 DESLIGA:   LD   A,#F0           3EF0
150            LD   (IY+15),A       FD770F
160 ACEAPA:    EX   (SP),IX         DDE3
170            PUSH IY              FDE5
180            JP   (IX)            DDE9
190 TEMPO:     LD   A,(#FCA2)       3AA2FC
200            XOR  #01             EE01
210            RET  NZ              C0
220 LIGA:      LD   (PISCA+1),A     3290FA
230            RET                  C9
240 ; Instruções para "Desativar" a rotina
      CURSOR PISCANTE
250 DESATIVA:  LD   A,#C9           3EC9
260            LD   (#FFC5),A       32C5FF
270            LD   (#FD9A),A       329AFD
280            LD   (#FDC2),A       32C2FD
290            RET                  C9
300 ; Instruções para "Ativar" a rotina
      CURSOR PISCANTE
310 ATIVA:     LD   HL,#FA75        2175FA
320            LD   (#FDC3),HL      22C3FD
330            LD   HL,#FA9C        219CFA
340            LD   (#FD9B),HL      229BFD
350            LD   HL,#FAA6        21A6FA
360            LD   (#FFC6),HL      22C6FF
370            LD   A,#C3           3EC3
380            LD   (#FFC5),A       32C5FF
390            LD   (#FD9A),A       329AFD
400            LD   (#FDC2),A       32C2FD
410 ; Instruções para inicializar os
      contadores de tempo (INTCNT/INTVAL)
420            LD   HL,#000A        210A00
430            LD   (#FCA0),HL      22A0FC
440            LD   (#FCA2),HL      22A2FC
450            RET                  C9
```

Execute o seguinte programa BASIC e procure digitar mais de 40 caracteres enquanto o desenho se forma na tela. Aguarde a sua conclusão e, depois, veja quais e quantos caracteres foram aceitos pelo MSX, mesmo enquanto o Z-80 estava ocupado com a tarefa de desenhar.

```
10 SCREEN2:FORI=0TO191:LINE(0,0)
-(255,I),IMOD15:NEXTI:SCREEN0
20 A$=INKEY$:IFA$=" "THENEND:
ELSE:PRINTA$;" ";:GOTO20
```

A instrução BASIC "A$=INKEY$" retira um caracter da Fila de Entrada do Teclado e o coloca na variável A$.

Perceba que a "repetição automática de teclas" não funciona nesta situação (só funciona se o BUFFER do Teclado for mantido vazio, com a retirada de cada caracter imediatamente após digitado).

Para mostrar as rotinas do sistema encarregadas de "examinar o teclado e carregar a fila" e de "retirar caracteres da fila", apresentaremos, em seguida, um programa que intercepta estas rotinas e faz com que o CURSOR, figura fundamental para a comunicação micro-usuário, se torne PISCANTE, o que se apresenta como um valioso recurso adicional para esta comunicação.

A rotina a seguir mostrada apresenta as seguintes características:

1) Utiliza as próprias rotinas do BIOS para "Mostrar" e para "Apagar" o CURSOR.

2) Intercepta a própria rotina do BIOS encarregada de "Retirar caracteres da Fila do Teclado" para implementar os procedimentos de "Mostrar" e "Apagar" o CURSOR.

Explicação detalhada do programa CURSOR PISCANTE MSX

*** As Instruções 20 a 60 estabelecem dados iniciais para a rotina Cursor Piscante.

*** 20

Instrui o compilador ASSEMBLER Z-80 para colocar o programa em linguagem de máquina (Programa Objeto) a partir do endereço #FA75).

Este endereço pertence ao BUFFER da Fila Musical – C do MSX.

*** 30

Coloca o endereço #09DA nos registradores HL do Z-80.

Este endereço corresponde ao início da rotina do BIOS encarregada de "Apresentar o CURSOR".

*** 40

Troca o conteúdo de HL com o conteúdo da última posição da "Pilha do Sistema". (Esta posição continha o endereço de retorno à chamada do Gancho #FDC2 pela rotina CHGET, pois é a partir deste Gancho que é implementado o desvio para #FA75.)

*** 50

Coloca o endereço #0A27 nos registradores IX do Z-80.

Este endereço corresponde ao início da rotina do BIOS encarregada de "Apagar o CURSOR".

*** 60

Carrega nos registradores IY do Z-80 o valor #FA81, correspondente ao endereço de memória da "Instrução número 70" do programa Cursor Piscante (Instrução CALL #0D6A).

*** As instruções 70 até 130 verificam se existe algum caracter no BUFFER do Teclado "aguardando" para ser retirado (a mesma coisa que seria feita pela rotina CHGET, caso não tivesse sido interceptada).

Em caso afirmativo, é efetuado o retorno para a rotina CHGET. Em caso negativo, ou seja, nenhuma tecla foi acionada pelo usuário, são executadas as instruções seguintes, encarregadas de "piscar" o CURSOR a intervalos regulares de tempo.

Neste ponto, o programa tem acesso a cada caracter digitado pelo usuário no Teclado, antes de ser processado pelo Interpretador BASIC ou outro programa que esteja ativo. VOCÊ PODE ADAPTAR ESTA ROTINA PARA DIVERSAS OUTRAS FINALIDADES – USE A IMAGINAÇÃO!

*** 70

Efetua chamada à rotina CHSNS (#0D6A) do BIOS, encarregada de "Verificar se existe caracter disponível no BUFFER do Teclado".

Em caso afirmativo, o "Indicador de Zero" do "Registrador de Status – F" é setado em "0" (NZ).

Se não há caracter disponível, este indicador é setado em "1" (Z).

*** 80

Caso não haja caracter disponível (JR Z), desvia para o endereço #FA8F (Instrução 130).

*** 90

Retira da Pilha do Sistema o último valor lá armazenado (#09DA ou #0A27).

Inicialmente, é colocado nesta posição da Pilha o valor #09DA (intruções 30/40), e a cada "piscada" do CURSOR este valor é alternadamente trocado com o valor #0A27 nos registradores IX (instruções 50/160).

3) Utiliza o próprio "Contador" interno do BIOS para determinar os intervalos de tempo para mostrar/apagar.

4) O programa é instalado nas posições de memória reservadas para armazenamento da "Fila Musical – C" do comando PLAY.

Com isto, consegue-se:

A) Uma rotina "simples" e confiável.

B) A função de "PISCAR O CURSOR" é ativada somente enquanto a rotina do BIOS encarregada de obter um caracter estiver funcionando, tornando-a plenamente compatível com o funcionamento global do MSX.

C) Funciona tanto em SCREEN0 como em SCREEN1, em qualquer situação.

D) O CURSOR permanecerá piscante para todos os programas que se utilizarem da rotina do BIOS.

Este é o caso do Interpretador BASIC, tanto para a situação em que "Aguarda comando do usuário" quanto para a situação em que atende a um comando "INPUT" de um programa BASIC.

É o caso, também, do Sistema Operacional DOS enquanto aguarda comando.

E) A velocidade de Acender/Apagar do CURSOR pode ser estabelecida pela utilização do comando "ON INTERVAL=nn" do BASIC.

F) Se for acionada a função musical PLAY do BASIC, o programa para CURSOR PISCANTE, automaticamente, se "Desliga", para evitar que ele possa ser destruído e ocasionar uma "Pane" do microcomputador. (Para desativar a rotina, baste teclar PLAY + RETURN).

São os seguintes os campos e rotina do BIOS utilizadas para tornar o CURSOR PISCANTE:

#0C3C – KEYINT

Rotina para processar "Interrupções" do Z-80.

Sempre que o Z-80 é interrompido pelo VDP (60 vezes por segundo), é efetuado um desvio para esta rotina. Esta rotina efetua uma "Chamada com retorno imediato" (Gancho) para o endereço #FD9A, o qual será utilizado para interceptá-la.

#10CB – CHGET

Rotina para recuperar o próximo caracter a partir da Fila do Teclado (BUFFER do Teclado), colocando-o

Após esta instrução, portanto, o Registrador A do Z-80 conterá o valor #09 ou #0A, indicando a última situação do CURSOR, se ele está "Apresentado" (#0A) ou "Apagado" (#09).

Esta instrução serve, ao mesmo tempo, para retirar da Pilha o valor lá colocado pela chamada ao Gancho em #FDC2.

*** 100

Compara o conteúdo do Registrador A com #09.

*** 110

Caso o Registrador A contenha #09, retorna à rotina CHGET no endereço #10E1, para dar continuidade àquela rotina.

*** 120

Caso o Registrador A não contenha #09, retorna à rotina CHGET no endereço #10DE, onde existe a instrução "CALL #0A27" que "Apaga o CURSOR", e segue para o endereço #10E1 para continuação dos procedimentos.

*** 130

Esta instrução de desvio é "Dupla", no sentido de que é "modificada" pela própria rotina para desviar para dois endereços diferentes.

Normalmente, ela contém "18F0", que corresponde à instrução Assembler "JR #FA81", ou seja, retorna à instrução número 70 para repetir a verificação do BUFFER do Teclado.

Em intervalos regulares de tempo, porém, o seu operando de endereço #F0 é substituído por #00 pela rotina encarregada de "contar o tempo". Com isto, esta instrução se transforma em "1800" (JR #FA91), passando a desviar para a instrução seguinte, pertencente à rotina encarregada de piscar o CURSOR.

*** As instruções 140 a 180 executam alternadamente as rotinas de "Mostrar" e de "Apagar" o CURSOR a cada nova passagem.

*** 140

Coloca o valor #F0 no Registrador A do Z-80.

*** 150

Armazena o valor #F0 na posição #FA90, correspondente ao operando de endereço da instrução número 130, retornando, portanto, esta instrução ao seu valor original "18F0" (JR #FA81).

Esta instrução, então "desliga" a rotina de Acender/Apagar o CURSOR.

*** 160

Esta instrução "troca" o conteúdo dos registradores IX com o conteúdo armazenado no topo da Pilha do Sistema.

Com este procedimento, os registradores IX conterão alternadamente os valores #09DA e #0A27 a cada nova passagem por esta instrução.

*** 170

Coloca na Pilha do Sistema o endereço #FA81, correspondente à instrução número 70 (CALL #0D6A).

*** 180

Desvia para o endereço contido nos Registradores IX do Z-80, que será #09DA ou #0A27, correspondentes às rotinas de "Mostrar" e de "Apagar" o CURSOR.

As instruções 170/180 equivalem a uma instrução CALL para os endereços mostrados, já que ao seu final será executado desvio (RET) para o endereço colocado no topo da Pilha do Sistema (#FA81).

*** As instruções 190 a 230 são acionadas a partir do Gancho em #FD9A, pertencente à rotina KEYINT do BIOS, que é acionada a intervalos regulares de tempo pelo Z-80, via "interrupções" geradas pelo VDP.

Estas instruções verificam se um certo intervalo de tempo decorreu, e em caso afirmativo "modificam" para #00 o operando de endereço da instrução número 130, convertendo-a em um desvio para a instrução seguinte (#FA91).

Com os procedimentos já descritos, aquela instrução será normalmente um desvio para #FA81, porém será um desvio para #FA91 uma vez a cada intervalo decorrido.

*** 190

Carrega no Registrador A o valor contido em #FCA2. Esta posição de memória corresponde ao "BYTE de mais baixa ordem" do campo INTCNT (Interruption Count – Contador Interrupções) do BIOS, no qual é automaticamente "subtraído 1" a cada passagem pela rotina KEYINT.

Quando INTCNT chega ao valor zero, ele é re-inicializado com o valor do campo INTVAL (Interval Value – Valor do Intervalo), guardado em #FCA0/FCA1, que pode ser alterado pela instrução "ONINTERVAL=nn" do BASIC.

Para alterar a freqüência na qual o CURSOR pisca, basta digitar ONINTERVAL=nn + RETURN.

no "Registrador-A" do Z-80.

Esta rotina efetua uma "Chamada com retorno imediato" (Gancho) para o endereço #FDC2, o qual será utilizado para interceptá-la.

#FCA2/#FCA3 – INTCNT

Campo utilizado pelo BIOS para "contar o tempo", verificando se o intervalo estabelecido pela instrução "ON INTERVAL=nn" já foi alcançado.

#FCA0/#FCA1 – INTVAL

Campo utilizado pelo BIOS/BASIC para guardar o intervalo de tempo estabelecido pela função "ONINTERVAL". ■

Para criar o programa CURSOR PISCANTE, você pode utilizar o seguinte programa BASIC:

(Para executá-lo, comande: BLOAD "CURSOR.ASS",R).

```
 10 DATA21,DA,09,E3,DD,21,27,0A
 20 DATAFD,21,81,FA,CD,6A,0D,28
 30 DATA09,F1,FE,09,CA,E1,10,C3
 40 DATADE,10,18,F0,3E,F0,FD,77
 50 DATA0F,DD,E3,FD,E5,DD,E9,3A
 60 DATAA2,FC,EE,01,C0,32,90,FA
 70 DATAC9,3E,C9,32,C5,FF,32,9A
 80 DATAFD,32,C2,FD,C9,21,75,FA
 90 DATA22,C3,FD,21,9C,FA,22,9B
100 DATAFD,21,A6,FA,22,C6,FF,3E
110 DATAC3,32,C5,FF,32,9A,FD,32
120 DATAC2,FD,21,0A,00,22,A0,FC
130 DATA22,A2,FC,C9
140 FORI=&HFA75TO&HFAD8:READA$:
    POKEI,VAL("&h"+A$)
150 BSAVE"CURSOR.ASS",&HFA75,
    &HFAD8,&HFAB2
```

Luiz Carlos Bittencourt é Engenheiro Eletricista, trabalha em Processamento de Dados desde 1971, e é autor da publicação BIT-BASIC (Software + Livro).

*** 200
Efetua uma operação XOR do Registrador A com o valor #01.
Esta operação resulta em "zero" somente se o registrador A contiver #01. (Portanto, sempre que o "contador" INTCNT chegar ao valor #01.)

*** 210
Se INTCNT não contém #01, retorna à rotina KEYINT sem qualquer ação.

*** 220
Sempre que INTCNT chega ao valor 01, coloca o valor contido no Registrador A (#00) no campo de endereço da instrução número 130, transformando-a em um desvio para a instrução 140.

*** 230
Retorna à rotina KEYINT.

*** As instruções 250 a 290 "desativam" a rotina CURSOR PISCANTE sempre que um comando PLAY é acionado, pois esta rotina está instalada em área utilizada por este comando. Isto é feito "desligando" os Ganchos das rotinas CHGET/KEYINT/PLAY.

*** 250
Coloca o valor #C9 (RET) no Registrador A.

*** 260
Coloca o valor #C9 no endereço #FFC5. (Desliga o Gancho da rotina PLAY.)

*** 270
Coloca o valor #C9 no endereço #FD9A. (Desliga o Gancho da rotina KEYINT.)

*** 280
Coloca o valor #C9 no endereço #FDC2. (Desliga o Gancho da rotina CHGET.)

*** 290
Retorna o controle para o sistema.

*** As instruções 310 a 400 "ativam" a rotina CURSOR PISCANTE instalada a partir do endereço #FA75, "ligando" os Ganchos das rotinas CHGET/KEYINT/PLAY.

*** 310
Coloca em HL o valor #FA75, correspondente ao endereço de início da rotina CURSOR PISCANTE.

*** 320
Coloca #FA75 em #FDC3/FDC4 (Gancho de CHGET).

*** 330
Coloca #FA9C em HL (rotina de Tempo do programa Cursor Piscante).

*** 340
Coloca #FA9C em #FD9B/FD9C (gancho de KEYINT).

*** 350
Coloca #FAA6 em HL (rotina de Desativação do programa Cursor Piscante).

*** 360
Coloca #FA9C em #FFC6/FFC7 (gancho de PLAY).

*** 370
Coloca o valor #C3 no Registrador A do Z-80.

*** 380
Coloca #C3 em #FFC5. Com isto, o Gancho em #FFC5 fica com #C3A6FA (JP #FAA6).

*** 390
Coloca #C3 em #FD9A. Com isto, o Gancho em #FD9A fica com #C39CFA (JP #FA9C).

*** 400
Coloca #C3 em #FDC2. Com isto, o Gancho em #FDC2 fica com #C375FA (JP #FA75).

*** As instruções 420 a 450 inicializam os contadores de tempo INTVAL e INTCNT.

*** 420
Coloca o valor 10 (#000A) nos Registradores HL.

*** 430
Coloca #000A (Dez) no campo INTVAL (#FCA0). Com isto, as instruções encarregadas de piscar o Cursor serão acionadas a cada dez passagens do sistema pela rotina KEYINT.

*** 440
Coloca o valor 10 no campo INTCNT (#FCA2), que é decrementado de uma unidade a cada passagem por KEYINT e substituído por INTVAL quando chega a zero.

*** 450
Retorna o controle para o Interpretador BASIC.

ASSEMBLER
BASIC

# CRIANDO LINHAS

*DIVINO C.R. LEITÃO*

Certamente você já se deparou com algum programa que, ao gerar uma tela, uma definição de sprite ou outros tipos de dados, o faz criando linhas em BASIC. O objetivo deste artigo é mostrar como isto pode ser feito.

Há várias formas de se chegar a este resultado, sendo que a forma aqui apresentada deriva de duas rotinas apresentadas no livro +50 Dicas, da editora Aleph. O que fiz foi passar para a linguagem de máquina a idéia dos programas 29 e 30 do citado livro.

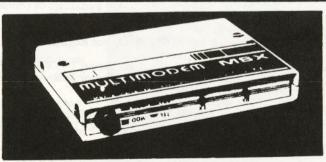
A excelente arquitetura do MSX permite um total domínio dos caminhos percorridos pelo sistema, mesmo quando se usa o BASIC. Em nosso caso específico, iremos interromper o processamento normal do buffer do teclado, fazendo com que o mesmo execute a tarefa de criar linhas de qualquer tipo.

O buffer do teclado do MSX tem um funcionamento interessante: dois

**figura 1**

```
C300   21 F0 FB E5 36 0B 23 36
C308   0D 23 36 0A 23 36 47 23
C310   36 4F 23 36 54 23 36 4F
C318   23 36 20 23 36 31 23 36
C320   30 23 36 0D 23 22 F8 F3
C328   E1 22 FA F3 C9 00 00 00
```

**figura 2**

```
1 CLEAR 500,&HC300
2 BLOAD"CRIALIN.BIN":'CARREGA O PROGRAMA
DA FIGURA 1
3 DEFUSR=&HC300
4 GOTO 20
10 GOTO 30
20 POKE &HC3FF,100
30 A=PEEK (&HC3FF):A=A+10:IF A=180  THEN
END
40 POKE &HC3FF,A
50 A$=STR$(A)+" DATA 00000000"
60 GOTO 9999
9999 CLS:LOCATE 1,0:PRINT A$:A=USR(0)
```

apontadores, chamados GETPNT (get pointer ou ponto de entrada) e PUTPNT (put pointer ou ponto de saída), percorrem uma área fixa de 40 bytes – iniciada em 64496 (&HFBF0) – como se fosse uma pista circular. Isto significa que, ao ultrapassar o valor do 40º byte desta área, qualquer um dos apontadores volta ao início da mesma, como em um loop eterno. Durante o processamento normal do MSX, estes apontadores ficam em constante movimento, obedecendo a uma regra simples: um não ultrapassa o outro.

Quando você pressiona uma tecla, uma das rotinas do BIOS verifica a sua validade – algumas teclas tais como SHIFT, ESC, CTRL, etc. não afetam o buffer do teclado – Caso a tecla seja válida, seu código será colocado na posição apontada por PUTPNT e o registro será incrementado de um byte. Caso PUTPNT alcance GETPNT, isto significará para o micro que o buffer está cheio. As teclas digitadas após esta situação serão ignoradas pelo interpretador.

Durante o processamento, o MSX tem que executar diversas tarefas e apenas em alguns intervalos é feito o esvaziamento do buffer do teclado, quando isto acontece os códigos apontados por GETPNT são processados por rotinas diversas, tais como um INPUT ou saída na tela. Nestes intervalos, GETPNT avança um byte após cada tecla processada. No entanto, o código da tecla não é apagado do buffer. O micro só não lê a sujeira deixada, porque, quando GETPNT alcança PUTPNT, isto irá indicar que o buffer foi esvaziado.

Como PUTPNT e GETPNT são variáveis na memória RAM, podem ser modificadas por um simples POKE, desta forma poderemos simular o acionamento do teclado. Esta é a idéia para auxiliar na criação de linhas. O que será jogado no buffer do teclado não será a linha propriamente dita, mas sim alguns códigos que irão simular o apertar da tecla RETURN.

**figura 3**

```
LD     HL,#FBF0    ;HL=INICIO DO BUFFER
PUSH HL            ;PRESERVA HL
LD     (HL),11     ;HOME (CURSOR EM 1,1)
INC    HL          ;INCREMENTA POSICAO
LD     (HL),13     ;RETURN
INC    HL
LD     (HL),10     ;CURSOR NA PROXIMA LINHA
INC    HL
LD     (HL),"G"    ;COMANDO GOTO 10
INC    HL
LD     (HL),"O"
INC    HL
LD     (HL),"T"
INC    HL
LD     (HL),"O"
INC    HL
LD     (HL)," "
INC    HL
LD     (HL),"1"
INC    HL
LD     (HL),"0"
INC    HL
LD     (HL),13     ;OUTRO RETURN
INC    HL
LD     (#F3F8),HL;PUTPNT = HL
POP    HL           ;RECUPERA INICIO DO
BUFFER
LD     (#F3FA),HL;GETPNT = INICIO DO
BUFFER
RET                ;VOLTA AO BASIC
```

**figura 4**

```
LINHA 0001: RESERVA AREA A PARTIR DE
            &HC300 PARA O PROGRAMA LM
            (PODE SER OUTRO ENDERECO)
LINHA 0002: CARREGA O PROGRAMA EM LM
LINHA 0003: DEFINE ENTRADA PARA A
            FUNCAO USR
LINHA 0010: ESTA E A LINHA DE RETORNO
            DA ROTINA LM, PODE CONTER
            QUALQUER COMANDO
LINHA 0020: PRESERVA O VALOR 100 EM UM
            ENDERECO DE MEMORIA
LINHA 0030: ATRIBUI VALOR A VARIAVEL
            "A", QUE SERVE PARA CHECAR
            FIM DO PROGRAMA E NUMERAR
            AS LINHAS A SER CRIADAS
LINHA 0040: PRESERVA VALOR DE "A"
LINHA 0050: ATRIBUI A STRING "A$" O
            CONTEUDO DA LINHA A SER
            CRIADA
LINHA 0060: CHAMA A ULTIMA LINHA DO
            PROGRAMA, QUE DEVE CONTER
            OBRIGATORIAMENTE OS MESMOS
            COMANDOS DA LISTAGEM.
            (A$ PODE SER SUBSTITUIDO)
LINHA 9999: ESTA LINHA SERA USADA PARA
            CRIAR AS NOVAS LINHAS DO
            SEU PROGRAMA EM BASIC.
            PODE TER QUALQUER NUMERO,
            MAS DEVERA SER SEMPRE A
            ULTIMA DO PROGRAMA.
```

Existe um problema ao criar uma linha durante o processamento de um programa: as variáveis criadas serão perdidas. A única forma de preservar valores importantes é salvá-los em uma área de memória livre através de um comando POKE. No programa exemplo (figura 2) as linhas 20 e 40 fazem justamente isto.

A rotina apresentada neste artigo, trabalha de forma simples. Ela simula o pressionar da tecla RETURN a partir da posição de vídeo 1,1 e, em seguida, simula o comando direto GOTO 10. Para utilizá-la em seu programa, você deverá imprimir o conteúdo completo da linha que deseja criar, inclusive o número da mesma – a partir da posição 1,1 do vídeo – e chamar a rotina com a função USR do BASIC. O controle do programa será devolvido para a linha 10. A existência da linha não será verificada e o formato da mesma deverá obedecer às regras do BASIC MSX.

A única exigência da rotina é que a linha que contém a função USR seja a última do programa e que a função seja o último comando da linha. Isto é necessário para que o interpretador BASIC não continue executando o programa e devolva o controle para a linha 10. No exemplo, a linha está com o número 9999 para destacá-la das outras, mas o número pode ser qualquer um, desde que seja a última linha do programa. A volta para a linha 10 foi definida aleatoriamente. Se quiser mudar para outra linha, altere o programa assembler nos locais onde ficam os bytes dos algarismos 1 e 0. Na figura 1, estes bytes ficam nos endereços &HC31D e &HC320. Veja a explicação dos dois programas para entender seu funcionamento.

A aplicação desta rotina é muito variada. No exemplo da figura 2, apenas cria 8 linhas DATA a partir de 100, mas pode ser usada para inúmeras outras atividades, como por exemplo a definição de fórmulas via INPUT, criar linhas contendo seqüências de comando DRAW a partir de movimentos do cursor e outras que você mesmo irá descobrir.

O objetivo deste artigo não foi simplesmente apresentar um programa, que em si não tem mérito algum, e sim tentar mostrar novas formas de usar as potencialidades escondidas de seu MSX.

Se você faz uso destes pequenos truques, envie-os para a CPU em forma de artigo. Seu trabalho será devidamente remunerado e também apreciado pelos seus colegas usuários.

A listagem da figura 1 contém os códigos em hexa da rotina em assembler que simula a digitação da linha. Coloque-a em qualquer endereço de memória, pois é uma rotina relocável. A figura 2 contém um exemplo de utilização, em BASIC. A figura 3 contém a mesma rotina da figura 1, em mnemônicos assembler e a figura 4 explica cada linha do programa em BASIC, para que você possa compreendê-lo melhor. ■

# ABASTEÇA O SEU MSX NA ECTRON.

**EQUIPAMENTOS PARA MSX**

Driver MSX 5 1/4
Driver MSX 3 1/2
Video Station
Interface p/ Drive
Cartão 80 Colunas
Modem
Monitores de Vídeo
Gabinete e Fonte p/ Driver
Ponta Discos "Acrílico" 100 Discos
Mesa para Computador
Mesa para Impressora

**— Fitas de Vídeo**

• Na Ecatron você encontra o último lançamento "MPO" em videocassete "Curso de Basic MSX" acompanha livro.
• "Dominando e MSX"

**SOFTWARE**

• D.Base Ferramenta Profissional p/ manipulação de banco de dados.
• Super Calc: A mais famosa Planilha de cálculos.
(Ambos com suporte técnico e reposição de versão)

**SUPRIMENTOS**

Disquetes
Fitas p/ Impressora
Formulários Contínuos
Capas Protetoras p/ Equipamentos

**APLICATIVOS**

Os mais potentes do mercado

**LITERATURAS**

Livros 100 Dicas
Livros 50 Dicas (EM LANÇAMENTO)
Livros Programação Avançada
Livros Astrologia
Livros Curso de Música
Livros Curso de Basic

**JOGOS**

Temos a coleção completa inclusive os últimos lançamentos.

Solicite nosso catálogo inteiramente GRÁTIS.

* Nossa caixa Postal 12005 - Cep. 02098/ São Paulo/

**ECTRON ELETRÔNICA LTDA.**

Rua Dr. Cesar, 131 - Metrô Santana - S.Paulo/SP

**TEL.: (011) 290-7266**

GTM&P

# PROGRAMA BIT BASIC

Ao receber para análise o livro e programa BIT-BASIC, minha atenção logo foi despertada para o ineditismo do material: um programa contendo um manual em forma de livro. Só havia visto tal procedimento antes em manuais de equipamentos, tais como CPU de micro, impressoras e, é claro, programas estrangeiros.

Após leitura atenta do livro e testes do programa, percebi que tinha em minhas mãos um tesouro. Ambos demonstraram uma qualidade a qual não estou acostumado. Que me perdoe o leitor pela euforia, mas estou tentando apenas repassar a vocês o mesmo entusiasmo por que fui tomado ao analisar este excelente material.

Cheguei a ficar frustrado, no bom sentido, pois não consegui sequer encontrar falhas no programa e os erros microscópicos que encontrei no manual nem compensam ser citados. Seria um desmerecimento ao trabalho do autor.

Mas o que é o BIT-BASIC? Modestamente, Luiz Carlos Bittencourt, o autor, apresenta-o como "soft que acrescenta novas funções ao BASIC". No entanto, o programa vai muito além. É um verdadeiro sistema que permite modificações personalizadas ao gosto do usuário.

Outro termo encontrado no manual, e que se encaixa como luva à proposta do mesmo, é "uma ponte para o assembler". Realmente, nunca o usuário do MSX BASIC esteve tão próximo da linguagem de máquina, além de apresentar conceitos de fácil compreensão, cada passo do programa BIT-BASIC é mostrado, comentado e explicado à exaustão. Os usuários ávidos por informações vão fazer uma festa.

É a primeira vez que um programa é aberto desta forma ao usuário. Nada fica obscuro. Cada instrução é explicada de forma objetiva e prática, com exemplos de sua aplicação. As explicações não se restringem ao BIT-BASIC, o BIOS e o BASIC do MSX, assim como o HOOKS e variáveis do sistema são dissecados a fundo. O livro, só pelo seu conteúdo técnico, merece estar na cabeceira do programador que realmente quer conhecer a fundo seu micro.

Ao ser carregado, o BIT-BASIC mostra uma tela que vale a pena ser reproduzida (veja figura), não pela sua qualidade gráfica, que é bastante simples, e sim pela sua mensagem de trabalho e honestidade para com o usuário. Vamos esperar que este corresponda.

Ao ser instalado, este programa não dá o conhecido e imbecil RESET, que a maioria dos pirateadores insiste em nos impingir, ficando residindo entre os endereços &H7000 e &H8000. A maneira como isto é feito já foi abordada em CPU número 2, página 15, e está muito bem explicada no manual do BIT-BASIC.

A seguir, serão enumeradas algumas das funções que considerei de maior auxílio ao programador do BASIC MSX.

Sem dúvida, a mais interessante é a possibilidade de se digitar comandos de forma simplificada, por exemplo: FILES vira .f e pode, inclusive, receber parâmetros, ou seja, FILES"B:*.BAS" pode ser trocado por .fb*.BAS ou BLOAD"CAS.prg",&HAACC,&HBBDD,&HFFFF fica simplesmente .blc.AACC,BBDD,FFFF. O mais importante, porém, não é a existência de comandos simplificados e sim a possibilidade de criá-los ao critério do usuário. Acompanha o BIT-BASIC um programa para a criação e modificação de uma tabela de comandos, cujo limite fica por conta da necessidade do usuário.

Dentre os novos comandos, destacaram-se o de cópia e movimentação de linhas, este último chegando ao requinte de renumerar os GOTOS e GOSUBS das linhas envolvidas, como o glorioso RENUM do BASIC padrão. O comando ".z", que permite a existência simultânea de dois programas BASIC, ou a união dos mesmos como no MERGE, mas sem destruir as linhas de números iguais, também é de extrema utilidade. É injusto tentar comparar um ou outro comando, pois todos são de enorme utilidade. Mas deve ser destacado que o usuário mais esperto, após a leitura do manual, poderá, tranqüilamente, criar seus próprios comandos.

Podem, também, ser vistos e editados, o conteúdo da memória. Pesquisa de strings dentro da linha BASIC, vira brincadeira de criança e a listagem dos programas poderá ser feita de diversas formas, para frente e para trás, sob total controle do usuário. Qualquer um destes recursos isolados já seria motivo mais que suficiente para se desejar o programa.

Você deve estar se perguntando: e o meu BASIC padrão? Quanto perderei para ganhar estas novidades? Eu mesmo fiz esta pergunta, descrente que estava de tanta vantagem e, surpreendentemente, constatei que o BASIC continuou inteirinho, com todos seus defeitos e qualidades, sem ser afetado de forma alguma pelo enxerto adquirido. É claro que rodar programas desconhecidos em assembler pode gerar problemas, mas programas de sua autoria poderão conviver tranqüilamente com o BIT-BASIC. Como já foi dito, cada rotina deste, por mínima que seja, está documentada e você poderá fazer com elas o que quiser e for capaz.

O disquete que foi enviado para análise continha, ainda, alguns programas em BASIC de excelente nível, um editor de SPRITES, bastante inteligente, que grava a tabela criada no formato binário.

**Conclusão**

Após o que foi descrito, só posso recomendar aos usuários que utilizam o BASIC que procurem adquirir o programa, sem esquecer que se trata de um produto realmente nacional e que, principalmente pela sua qualidade, deve ser respeitado.

Ao autor e à MICROBIT Informática, que comercializa o soft, fica o elogio pelo bom trabalho realizado e a esperança de que este soft seja apenas o primeiro de uma série.

*SÉRGIO DURIC CALHEIROS*

# SCREEN IV

## PARTE 2

Dando prosseguimento ao projeto SCREEN IV, neste mês, faremos a implementação do módulo de impressão. Este módulo contém as rotinas que geram o texto em 64 colunas na tela gráfica, ativada pelo comando SCREEN 4.

Antes de mais nada, é necessário que a tabela de caracteres já esteja disponível, ou seja, que já tenha sido digitada e colocada em forma de arquivo em disco. Para que o leitor tivesse condições de digitá-la sem pressa, a tabela foi publicada na primeira parte do projeto SCREEN IV, no mês anterior. Com a tabela de caracteres pronta, podemos iniciar a implementação do módulo.

Usando o MSXDEBUG, prepare a página 1 para digitação. Este procedimento já pode ser considerado rotineiro para quem digitou o MSXDEBUG ou começou a digitar o SCREEN IV. Entretanto, para quem ainda não está familiarizado, basta preencher o trecho de memória de 4000H a 7FFFH com o byte 00H. Para isso, use o comando FILL 4000 7FFF 0.

Coloque no drive o disco com a primeira parte do projeto SCREEN IV e carregue o arquivo SCREEN.COM no endereço 4100H (DLOAD SCREEN.COM 4100). Em seguida, carregue a tabela de caracteres no endereço 4386H (DLOAD SET.ALF 4386).

Agora basta digitar o bloco 1 a partir do endereço 4B86H. Use o comando DISP 4B86.

Se o seu MSXDEBUG já for a versão 1.1, ou seja, já possuir o comando SOMA, você já pode conferir se algum BYTE foi mal digitado ou perdido. Some os BYTES do endereço 4B86H ao 4EF6H com o comando SOMA 4B86 4EF6. O valor lido deve ser o fornecido ao final do bloco 1. Se o valor não conferir, revise o texto e corrija. O comando SOMA está sendo implementado no MSXDEBUG neste número da revista CPU.

Com o bloco em ordem, devemos, agora, definir as chamadas das rotinas do módulo, isto é, fazer com que as rotinas de gerenciamento reconheçam e habilitem seu funcionamento.

Não creio que entrar em detalhes de como este reconhecimento é feito seja útil para o usuário do SCREEN IV. Mas, para que isso não se torne um mistério, basta ler os artigos do projeto MSXDEBUG. A técnica usada é a manipulação de endereços em tabela, de maneira semelhante à usada no MSXDEBUG.

As mudanças no bloco original do SCREEN.COM se resumem em acrescentar o endereço de entrada da rotina principal do módulo (@CHPUT) na tabela de rotinas e acrescentar o endereço do gancho (@HCHPU ) na tabela de ganchos. Estes endereços são 0B86H e FDA4H, respectivamente.

No endereço 4109H, digite os BYTES 086H e 0BH. No endereço 4168H, defina o gancho usado, digitando os BYTES 0A4H e 0FDH. Os endereços são colocados ao contrário, com a parte menos significativa na parte de memória mais baixa.

Agora, resta salvar o bloco único do SCREEN IV com o comando DSAVE SCREEN.COM 4100 4EF6.

A partir do DOS, execute o SCREEN.COM.

Além da mensagem normal do BASIC, nada mais deve ser notado. Para se convencer de que tudo está certo, digite as rotinas das listagens abaixo, que são programas de teste, escritos em BASIC estendido.

Muitos usuários do micro MSX, não possuem monitores de vídeo, usando apenas a televisão como monitor. Em algumas televisões, a imagem do canto esquerdo desaparece atrás da borda. Para que a imagem não desapareça, ao ativar o SCREEN IV pela primeira vez, teremos apenas 60 colunas, e não 64. No próximo mês, faremos a adaptação do comando WIDTH ao BASIC estendido. Por enquanto, se utilizado, o comando WIDTH só aceitará 40 colunas no máximo, como funciona no modo normal. Após a extensão do comando WIDTH, chegaremos às 64 colunas. Como recurso temporário, podemos mudar o número de colunas diretamente na própria variável do sistema que contém este número. Para isso, basta usar o comando do BASIC: POKE &HF3B0,NC onde NC é número de colunas. Um

número maior que 64 poderá causar distúrbios na tela, mas não deverá ser suficiente para causar uma queda do sistema. Use e abuse.

A partir de agora, deixarei que o próprio leitor se encarregue de descobrir as novas características e aplicações do módulo implementado. Acho que não preciso dizer mais nada; apenas deixar que a imaginação e necessidade de cada um o faça por mim.

Mês que vem, tem mais. Até lá! ■

LISTAGEM 1

```
10 REM Testa impressão do set de caracteres
20 SCREEN 4:REM Habilita tela gráfica e texto em 64 colunas
30 FOR F=32 TO 254
40 PRINT CHR$(F);" ";
50 NEXT
90 A$=INPUT$(1)
```

LISTAGEM 2

```
10 REM Testa comando LOCATE e COLOR em SCREEN 4
20 SCREEN 4
30 FOR F=0 TO 15
40 LOCATE F,F:COLOR F:PRINT "Posição:";F;",";F
50 NEXT
60 COLOR 15,1,1
90 A$=INPUT$(1)
```

LISTAGEM 3

```
10 REM Testa rolamento de tela
20 SCREEN 4
30 FOR F=0 TO 30
40 PRINT "Linha.";F
50 NEXT
60 LOCATE 0,0:PRINT "Digite uma tecla!":A$=INPUT$(1):LOCATE 4,4
70 PRINT CHR$(27);"L";:REM Insere linha n posição do cursor
80 PRINT CHR$(27);"M";:REM Deleta linha na posição do cursor
90 GOTO 70:REM Pare com CRTL+STOP
```

LISTAGEM 4

```
10 REM Testa funções de tela
20 SCREEN 4
30 FOR F=0 TO 20:PRINT TAB(F);"Função de tela":NEXT
40 LOCATE 0,0:PRINT "Digite uma tecla!":A$=INPUT$(1)
50 LOCATE 0,0:PRINT CHR$(27);"K";:REM Apaga até fim da linha
60 PRINT "Digite uma tecla!":A$=INPUT$(1)
70 PRINT CHR$(27);"J";:REM Apaga até fim da tela
80 PRINT "Digite uma tecla!"
90 A$=INPUT$(1)
```

LISTAGEM 5

```
10 REM Combina texto e gráfico
20 SCREEN 4
30 LINE (0,0)-(255,191),,B
40 LINE (2,2)-(253,189),,B
50 LOCATE 1,1
60 PRINT "Janela gráfica calculada!"
90 A$=INPUT$(1)
```

Bloco 1

```
4B86 C3 01 0D F7 00 C0 00 C9
4B8E 3A 63 38 11 00 01 21 00
4B96 00 47 A7 04 19 10 FD ED
4B9E 52 3A 64 38 11 08 00 CB
4BA6 3F 3C A7 47 ED 52 19 10
4BAE FD 3A 64 38 CB 27 CB 27
4BB6 E6 07 32 66 38 22 5F 38
4BBE CB 2C CB 1D CB 2C CB 1D
4BC6 A7 28 01 23 11 00 30 19
4BCE 22 67 38 C9 2A 5F 38 11
4BD6 00 38 CD 96 02 06 08 DB
4BDE 98 12 13 10 FA ED 5B 61
4BE6 38 21 00 38 06 08 3A 66
4BEE 38 A7 DD 21 6A 38 20 3A
4BF6 DD 21 6B 38 1A E6 F0 4F
4BFE 7E DD A6 00 B1 77 13 23
4C06 10 F2 2A 5F 38 CD 8A 02
4C0E 21 00 38 06 08 7E D3 98
4C16 23 10 FA CD 04 03 2A 5F
4C1E 38 11 00 20 19 CD 8A 02
4C26 06 08 3A 4A 38 D3 98 10
4C2E FC C9 1A E6 F0 CB 3F CB
4C36 3F CB 3F CB 3F 4F 7E DD
4C3E A6 00 B1 77 13 23 10 EA
4C46 18 C0 3A B0 F3 47 3E 41
4C4E 90 CB 3F 25 84 67 2D 22
4C56 63 38 C9 F7 00 32 0C C9
4C5E 3A 65 38 47 21 86 03 11
4C66 08 00 04 19 10 FD ED 52
4C6E 22 61 38 C9 F3 E5 26 01
4C76 3E 40 CD 4B 0C CD 8E 0B
4C7E 01 00 01 2A 5F 38 11 00
4C86 36 CD C3 0C 11 00 38 2A
4C8E 67 38 01 40 00 ED B0 E1
4C96 FB C9 F3 E5 26 01 3E 40
4C9E CD 4B 0C CD 8E 0B 3A B0
4CA6 F3 01 00 01 2A 5F 38 11
4CAE 00 36 EB CD D4 0C ED 5B
4CB6 67 38 21 00 38 01 40 00
4CBE ED B0 E1 FB C9 CD 96 02
4CC6 EB 51 0E 98 58 42 ED A2
4CCE 20 FC 43 10 F7 C9 EB CD
4CD6 8A 02 EB 51 0E 98 58 42
4CDE ED A3 20 FC 43 10 F7 C9
4CE6 F3 79 E5 F5 32 65 38 CD
4CEE 48 0C CD 8E 0B CD 5E 0C
4CF6 CD D2 0B 2A 67 38 F1 77
4CFE E1 FB C9 F5 F7 00 2E 0A
4D06 F1 00 CD 1B 0D F7 00 E1
4D0E 09 3A DD F3 3D 32 61 F6
4D16 DD 21 DA 08 C9 F7 00 9D
4D1E 08 D0 4F 20 0D 21 A7 FC
4D26 7E A7 C2 D0 0D 79 FE 20
4D2E 38 25 2A DC F3 FE 7F CA
4D36 C6 0E CD E6 0C CD 25 0E
4D3E C0 AF F7 00 2B 0C 26 01
4D46 CD 42 0E C0 CD 4A 0E 2E
4D4E 01 CD F2 0E C3 69 0E 21
4D56 6E 0D 0E 0C 23 23 A7 0D
4D5E F8 BE 23 20 F7 4E 23 46
4D66 2A DC F3 CD 6E 0D AF C9
4D6E C5 C9 07 89 0B 08 2D 0E
4D76 09 52 0E 0A 46 0D 0B 60
4D7E 0E 0C C6 02 0D 62 0E 1B
4D86 CA 0D 1C 3C 0E 1D 2D 0E
4D8E 1E 38 0E 1F 42 0E 6A C6
4D96 02 45 C6 02 4B D1 0E 4A
4D9E E2 0E 6C CF 0E 4C 96 0E
4DA6 4D 66 0E 59 C7 0D 41 38
4DAE 0E 42 42 0E 43 25 0E 44
4DB6 36 0E 48 60 0E 78 C1 0D
4DBE 79 C4 0D 3E 01 01 3E 02
4DC6 01 3E 04 01 3E FF 32 A7
4DCE FC C9 F2 DE 0D 36 00 79
4DD6 21 92 0D 0E 0F C3 5A 0D
4DDE 3D 28 1E 3D 28 25 3D 77
4DE6 3A B0 F3 11 DD F3 28 06
4DEE 36 03 CD 59 0C 1B 47 79
4DF6 D6 20 B8 3C 12 D8 78 12
4DFE C9 77 79 D6 34 28 0B 3D
4E06 28 0F C9 77 79 D6 34 20
4E0E 05 3C 32 AA FC C9 3D C0
4E16 3C 32 A9 FC C9 3A A9 FC
4E1E A7 C0 F7 00 E6 09 C9 3A
4E26 B0 F3 BC C8 24 18 1D CD
4E2E 36 0E C0 3A B0 F3 67 11
4E36 25 3E 2D C8 18 0E CD 25
4E3E 0E C0 26 01 CD 59 0C BD
4E46 C8 38 05 2C 22 DC F3 C9
4E4E 2D AF 18 F8 3E 20 CD 1B
4E56 0D 3A DD F3 3D E6 07 20
4E5E F3 C9 2E 01 26 01 18 E4
4E66 CD 62 0E CD 59 0C 95 D8
4E6E CA CF 0E E5 F5 4F 06 00
4E76 F7 00 1D 0C 6B 62 23 ED
4E7E B0 21 CA FB 35 F1 E1 F5
4E86 2C CD 72 0C 2D CD 98 0C
4E8E 2C F1 3D 20 F2 C3 CF 0E
4E96 CD 62 0E CD 59 0C 67 95
4E9E D8 CA CF 0E 6C E5 F5 4F
4EA6 06 00 F7 00 1D 0C 6B 62
4EAE E5 2B ED B8 E1 74 F1 E1
4EB6 F5 2D CD 72 0C 2C CD 98
4EBE 0C 2D F1 3D 20 F2 18 09
4EC6 CD 25 0E C8 0E 20 C3 E6
4ECE 0C 26 01 F7 00 29 0C 0E
4ED6 20 CD E6 0C 24 3A B0 F3
4EDE BC 30 F4 C9 E5 CD CF 0E
4EE6 E1 CD 59 0C BD D8 C8 26
4EEE 01 2C 18 F0 C9 C9 C9 C9
4EF6 C9 00 00 00 00 00 00 00
4EFE 00 00 00 00 00 00 00 00

Soma total:016108
```

# A NOSSA LINHA AGORA ANDA...

CBBS CPU

**CHEIA DE PROGRAMAS E INFORMAÇÃO.
24H POR DIA À VELOCIDADE DE 300/300.
ENTRE NESTE SISTEMA E FALE COM A CPU.
DISQUE (021) - 259-4689.
CPU RESPONDE.**

# CPU CARTAS

■ Parabéns pelo sucesso da revista CPU. Realmente, foi além das minhas expectativas.

Recebi um exemplar da revista com um pouco de atraso (esperava que vocês enviassem tão logo a minha carta chegasse até vocês).

Assim que voltei da viagem de férias, recebi a revista, tendo ficado muito interessado e apresentei-a aos sócios do clube CENTRO MSX, que dirijo aqui em Recife. Muitos gostaram e já devem ter feito a assinatura da CPU. É isso que estou fazendo agora.

Comprei o número 07 e fiquei surpreso com o avanço e com os elogios. Faço, agora, uma assinatura de 6 exemplares, a partir do número 8. Mando, em anexo, as informações solicitadas.

Gostaria que na seção de Cartas fosse impresso que desejo trocar dicas em Basic e Assembler com qualquer interessado. Peço também aos fanáticos por jogos que, se puderem, mandem informações de como jogar o simulador F-16, pois não consigo manuseá-lo.

Para os Adventure-maníacos, informo que está sendo formado um novo clube. Os interessados devem escrever para o seguinte endereço: SHIN QL 8 conjunto 8 casa 1 – Lago Norte – Brasília – DF – 71500, aos cuidados de Daniel C. Sobral.

Diógenes Souza Leão Filho
Rua Santa Lúcia, 65/402
54000 – Candeias – PE

■ Gostaria que vocês publicassem as senhas do jogo Goonies, em inglês, se possível, e um catalogador de fitas que imprimisse o endereço inicial, final e de execução dos programas em linguagem de máquina.

Fábio Villaricencio
Rua Goiânia, 238 – B. Botafcgo
95700 – Bento Gonçalves – RS

***O catalogador de fitas que você solicita em sua carta já foi publicado na revista CPU número 2 (Copiador de disco/fita e fita/disco).***

***O programa Goonies será analisado nas próximas edições da revista.***

■ Estou lhes escrevendo porque, recentemente, através de um amigo que mora em São Paulo, vim a adquirir um exemplar da revista CPU, onde, após ver as inovações que ela trouxe na área de informação ao usuário de MSX, fiquei sabendo da existência de vossa Softhouse. Constatei que tinha adquirido não apenas mais uma revista de informática, mas sim uma obra-prima indispensável ao usuário interessado em tirar o máximo proveito de sua poderosa máquina, que é este maravilhoso microcomputador doméstico, que se testa, atualmente, em outras áreas.

Mas qual não foi minha desilusão ao chegar ao jornaleiro, pedir a revista, e ser informado que nem conhecia tal publicação.

Por isso peço-lhes, encarecidamente, que me mandem informações, bem como o preço atual da assinatura da revista, para que venha a recebê-la em minha casa.

Se possível, enviem também um catálogo de software da Águia Informática, para que também venha a desfrutar de vossos préstimos como softhouse.

Certo de vossa compreensão, desde já agradeço.

Marcos Roberto Santos Vieira
Rua Guilherme Alves, 546/602
90630 – Porto Alegre – RS

***Atualmente, a quantidade de revistas que estamos enviando para Porto Alegre não é suficiente para cobrir todas as bancas, motivo pelo qual você não encontrou CPU em seu jornaleiro. Em breve, já estaremos enviando quantidade suficiente de revistas para poder atender a todas as bancas de sua cidade.***

***Com relação à venda de programas para MSX, informamos que não estamos mais atuando nesta área desde janeiro passado.***

■ Conheci a revista no número 7 e, por achar que a revista é imensamente útil para os usuários de MSX, assinei-a desde então. Gostaria de parabenizá-los pela sua qualidade.

O exemplar número 8 está simplesmente sensacional, mas me decepcionei ao digitar

o programa de conversão de telas para .SCR e perceber que ele não funciona. Ao carregar a tela, o computador age como se recebesse um RESET.

Há alguma modificação a ser feita? Caso haja, como faço para listar o programa, já que ele é gravado em formato binário?

Acabei de adquirir um drive e, como fã do Graphos III, acho esse programa absolutamente necessário. Corrijam-no, por favor. Talvez eu tenha errado ao digitá-lo. Nesse caso, como listá-lo?

Por ter a revista apenas a partir do número 7, gostaria de adquirir os números que perdi. Seria possível? Como proceder?

Gostaria de sugerir que vocês dessem dicas sobre os jogos Taipan e Triple Comand, principalmente o primeiro.

Por favor, ajudem-me com vossa experiência a resolver minhas dúvidas.

Parabenizo-vos mais uma vez.

Leonardo L. O. Santos
Av. Prudente de Morais, 1965/705
30380 – Belo Horizonte – MG

***Não verificamos erro no programa de conversão de telas gráficas. Acreditamos que tenha ocorrido algum erro na digitação do programa, principalmente nas linhas DATA, que contêm o programa em Assembler.***

***Sempre que for digitar um programa em Assembler, é aconselhável que, antes de executar o programa, faça uma gravação do mesmo, em Basic, pois lhe será muito mais fácil conferir o que foi digitado e fazer alterações, caso sejam necessárias.***

***Os números 1, 2 e 3 de CPU encontram-se esgotados e deverão ser reeditados em breve. Para adquirir os outros números (de 4 a 7), basta enviar um cheque nominal à ÁGUIA INFORMÁTICA, ou vale postal, sendo que cada número atrasado tem o mesmo custo do exemplar que é vendido em banca.***

■ Troco jogos de computador, em disco e fita. Também gostaria de trocar correspondência com todos os que possuem MSX.

Cláudio M. Neis
Caixa Postal 38
93900 – Ivoti – RS

■ Sou leitor assíduo de CPU desde o primeiro número e venho pedir um socorro a vocês com relação ao MSXDEBUG.

Digitei a parte Basic do programa, segui todas as instruções da parte II e, quando tentei usar o MSXDEBUG para digitar o SCREEN IV, me dei mal. O FILL, DIR, DSAVE, DLOAD e o DOS funcionam, mas o DISP, o EXEC e o MOVE não. Inclusive, quando usei o DISP 4100 'CR', apareceu a mensagem CD127Ú(4 Q00 e o cursor ficou duas linhas acima da linha SIS ou, às vezes, na última posição da linha. Qualquer coisa que fosse digitada fazia o cursor subir duas linhas, repetindo a mesma mensagem.

Pergunto: há possibilidade de erro nas linhas DATA, de forma que a soma dê certo (tipo trocar B por 8 ou vice-versa)? Também, quando procurei (com o DISP) o byte OFF, no endereço 4CFA, para inserir o comando SOMA, o equipamento travou todo o teclado. Nem ESC deu jeito.

Possuo apenas um montador .Im (do tipo que saiu na revista Input número 5, página 93) ou você recomenda um montador específico para o Screen IV?

Gostaria de entrar em contato com quem tenha o COMP32 (compactador de textos), pois perdi a parte Basic e gostaria de comprar o Multicalc para o TK2000 Color com manual, pois o meu não possui letras minúsculas no vídeo e nem saída para impressora.

Jorge Renato Alves da Silva
Rua Inválidos, 138/B1 2/COB
20231 – Rio de Janeiro – RJ

■ Após digitar toda a listagem do programa MSXDEBUG, publicado em CPU número 6, dei o comando que executasse o programa.

A execução teve início, mas, quando a contagem chegou a 196, a mesma foi interrompida com a mensagem de erro "OVERFLOW IN 200".

Depois de conferir se não havia cometido algum erro de digitação, tentei, novamente, executar o programa e a mesma interrupção ocorreu.

Decidi, então, escrever-lhes para que, se possível, me auxiliem.

João Carlos Sá
Rua Bras Sangiovanni, 23 ap 72
04304 – São Paulo – SP

***Em resposta a dúvidas que recebi por intermédio de cartas em relação ao projeto MSXDEBUG, posso afirmar, desde já, que todas as suas rotinas foram testadas e funcionam satisfatoriamente. Outro fato a ser considerado é a maneira como as listagens são geradas. Neste processo não há sequer um passo que esteja sujeito à falha humana. Todas as listagens são criadas pelo próprio microcomputador, a partir dos arquivos originais em disco. Após cada listagem ser gerada, há novamente um teste com os programas.***

***Em face disto, a possível causa dos erros que alguns leitores me mandaram pode ser, e deve ser, erro na digitação das listagens.***

***Ao leitor João Carlos Sá, recomendo uma revisão nos detalhes do programa, especialmente na parte montadora. Preste atenção nos caracteres pouco usuais, como o # e no %. Muitos leitores não tiveram problemas deste tipo e conseguiram fazer a montagem do MSXDEBUG.***

***Para o leitor Jorge Renato Alves da Silva, supondo que tenha passado pela parte montadora sem problemas, devo sugerir que a verificação do problema deva ser feita diretamente no código. De acordo com sua carta, os problemas se encontrariam nas rotinas dos comandos MOVE, DISP e EXEC. O comando MOVE se encontra do endereço 4122H a 4165H. Caso não seja encontrada nenhuma diferença de valores, verifique a sua chamada na tabela da rotina @INSTR. Isto será explicado detalhadamente na parte 2 do MSXDEBUG. Considerando que os demais comandos funcionaram corretamente, as rotinas de suporte ao MOVE também devem estar corretas. Finalmente, faça as mesmas verificações com o DISP e com o EXEC, que estão nos endereços 4177H a 45AEH e 4166H a 4172H, respectivamente. Isto deverá sanar quaisquer possíveis problemas.***

***Sérgio Duric Calheiros***

■ Possuo um Expert 1.1 e gostei muito da revista CPU, principalmente das seções de programas.

Dando uma sugestão, gostaria que vocês colocassem uma seção de programas comerciais e acho que o MSX não deve ser encarado como um vídeo game sofisticado, como muitos acham e querem que seja.

O MSX possui muitos

recursos que não são aproveitados. Não existem programas no mercado, apenas jogos.

Espero que a revista continue com a mesma qualidade que vem apresentando.

Douglas B. de Alcantara
Av. Higienópolis, 767
09790 – Santo André – SP

***Já existem muitos programas para uso profissional, desenvolvidos por várias softhouses, tanto do Rio de Janeiro como de São Paulo, que exploram os recursos do MSX, que já possui software nacional para ajudar os hobbystas de eletrônica (EDTRONIC), um Page Maker, da Nemesis Informática e muitos outros.***

***Nós, da revista, temos notado que os lançamentos de softwares nacionais têm sido constantes e que os programas são de bom nível, fazendo aquilo o que realmente é anunciado. Acompanhe a seção de análise de software.***

■ Adorei o artigo sobre o MSX 2.0 por transformação, publicado em CPU número 3, por isso, gostaria que em vosso próximo número nos mostrassem os MSX 2.0+ (plus), recentemente lançados no Japão. Dizem que os seus recursos, principalmente a resolução gráfica, é em muito superior à versão anterior. Gostaria, também, que apresentassem o mapa e dicas para o jogo Spelunker da Brotherbund Software e do Star Soldier.

Em CPU número 6 foram apresentados alguns comandos do MSX 2, mas de maneira muito vaga. Peço-lhes que nos mostrem todas as possibilidades de novos comandos, assim como das instruções que foram ampliadas, como a instrução SCREEN, por exemplo.

Parabéns pelo sucesso da revista e espero que aumentem a tiragem de cópias.

Marcos Jacoby
Caixa Postal 1058
Scharlau
93121 – São Leopoldo – RS

***Sempre que for possível, abordaremos o MSX 2. Neste número, na entrevista dada pelo Prof. Pierluigi, foi comentada alguma coisa sobre o MSX 2 e a transformação que está sendo realizada no Brasil.***

***Com relação aos comandos, estamos preparando uma série de artigos que irão abranger a totalidade dos novos comandos disponíveis no MSX 2.***

■ Gostaria de saber quais as alterações que teriam que ser feitas para que eu pudesse usar as dicas publicadas na revista para os jogos Colt 36 e Twin Bee, pois estes jogos que possuo têm um número de blocos diferentes.

Roberto Candal da Silveira
Rua Coronel Fernando Machado, 657 – Centro
90010 – Porto Alegre – RS

***Quando publicamos dicas de mil vidas, energia infinita ou plenos poderes na revista, testamos todos os POKE's. Acontece que algumas softhouses efetuam mudanças nos jogos, introduzindo algumas modificações. Nestes casos, não tendo o jogo para análise, fica inviável saber se os endereços foram alterado , fazendo com que os POKE's fornecidos não mais funcionem.*** ■

# DAWN PATROL

*ANDRÉ LUIS ANCIÃES DOS SANTOS*
*EDUARDO RIBEIRO TAVARES*
*MARCOS RIBEIRO TAVARES*

O jogo Dawn Patrol consiste em comandar um submarino pelo mar Mediterrâneo, com o objetivo de completar o maior número de missões possíveis. Diversos navios inimigos surgirão para destruí-lo. Você poderá combater ou fugir, já que certas missões exigem um tempo determinado para o seu complemento.

Você possui, no seu submarino, tubos de torpedos, motores a diesel e elétrico, periscópio e tudo mais que um submarino possui.

INSTRUMENTOS:

[1] Periscópio: através dele você tem visão de tudo o que está acontecendo à sua volta. Para girar o periscópio, basta pressionar a barra de espaços mais a seta direita ou esquerda. Quando estiver submerso, suba o periscópio, pressionando a seta para cima. Para descer, pressione a seta para baixo. Verifique se o submarino está em profundidade de telescópio, que é o segundo tracinho do marcador de profundidade.

[2] Marcador de velocidade: é onde você controla a potência dos motores ou a direção (à frente ou ré). Para aumentar a velocidade, digite "CONTROL" e, para diminuir ou retroceder, tecle "SHIFT". Seu submarino possui dois motores: um elétrico e outro a diesel. Quando submerso, você deverá utilizar o motor elétrico, pressionando a tecla "E" e, quando estiver na superfície, utilize o motor a diesel através da tecla "D".

[3] Leme: através das teclas de cursor (setas esquerda e direita), você controla uma bolinha vermelha que fará com que a seta se desloque, mudando, assim, a rota do submarino. Para parar o leme, basta retornar a bolinha para a posição norte do mostrador.

[4] Medidor de profundidade: por este medidor você controla a submersão do submarino. A bolinha vermelha indica a profundidade. Por isso, não deixe que a seta do mostrador se aproxime da bolinha, pois, caso isto ocorra, o submarino irá se chocar com o fundo do mar. Para submergir ou emergir, pressione as teclas do cursor para cima ou para baixo, respectivamente.

[5] Reservatório de Oxigênio: quando o submarino está submerso, o oxigênio vai sendo consumido. Para repor o oxigênio, é necessário retornar à superfície e pressionar a tecla "O". Assim que o tanque estiver cheio, pressione novamente a tecla "O".

O jogo ainda possui outros comandos, através das teclas de função F1, F2, F3, F4 e F5.

F1 – Casa das Máquinas: é na casa das máquinas que você observa as condições dos motores, além de verificar a quantidade de água no lastro. O lastro é um tanque que se enche e esvazia de água para submergir mais rapidamente, geralmente utilizado em emergências. Para encher o lastro, tecle "INSERT" e, para esvaziar, tecle "DELETE". A tecla "HOME" interrompe o enchimento e o esvaziamento do tanque. É recomendável deixar o lastro no nível 5.

F2 – Sala de Torpedos: seu submarino dispõe de 4 tubos de torpedos rontais e 2 de ré. Para lançar os torpedos, pressione "ESPAÇO" mais o número correspondente ao tubo de torpedo. Os números 1, 2, 3 e 4 são os tubos dianteiros e 5 e 6 são os traseiros. O submarino possui, ainda, 2 miras: uma vermelha, que vem a ser a mira frontal e uma branca, que é a mira traseira. É necessário girar o periscópio para se obter visão através da mira traseira. Para recarregar os tubos, tecle "L" mais o número do tubo desejado.

F3 – Mapa: é através do mapa que você tem a noção da direção de tempo nessa tela, principalmente se estiver submerso ou perseguindo algum navio.

F4 – Relatórios de Danos: indica todos os danos do submarino. Se, por acaso, o leme for destruído, seu submarino ficará à deriva. Se os danos forem muitos e resolver desistir da partida, tecle "STOP + CONTROL", para abortar o jogo.

F5 – Missões: nesta tela surgirá sua missão e, na parte inferior, suas coordenadas e um relógio. A todo o instante, novas informações a respeito de suas missões irão surgir nesta tela.

DICAS

Utilize somente o motor elétrico quando submerso. Observe a quantidade de energia na casa de máquinas. Quando estiver na superfície, use o motor diesel, pois, além do submarino de deslocar com mais velocidade, a energia do motor elétrico vai sendo recarregada.

Assim que completar alguma missão, você terá que voltar para a base. Lá chegando, desligue os motores e tecle "F", para reparar os danos e se reabastecer de torpedos e combustível.

Antes de submergir, verifique a profundidade local, através da bolinha vermelha do marcador de profundidade. Evite chegar próximo à costa, pois são locais muito rasos e, em caso de encontrar algum inimigo, não poderá submergir com segurança.

Certas missões possuem um tempo determinado para serem concluídas. Portanto, não se descuide do relógio da tecla F5. Se demorar demais, perderá o comando.

O submarino manobrará com mais facilidade submerso. Por isso, é recomendável que, em batalha, você esteja em profundidade de periscópio.

Se, ao submergir, tocar uma sirene, tecle "O", para fechar o compartimento de ar que por acaso ficou aberto quando recarregou o reservatório de ar.

Assim que o torpedo for lançado, surgirá, na parte inferior da tela a mensagem "torpedo launched", avisando seu lançamento. Quando for muito grande o número de navios a sua volta, fuja e não tente lutar. ■

# ABADIA DEL CRIMEM

*ANDRÉ LUIS ANCIÃES DOS SANTOS*
*EDUARDO RIBEIRO TAVARES*
*MARCOS RIBEIRO TAVARES*

Neste jogo, que foi inspirado no filme "O Nome da Rosa" você será um Abade chamado Frei Guillermo. Seu objetivo é descobrir quem está por trás dos assassinatos que vêm acontecendo.

A conclusão do jogo é muito difícil. O máximo que conseguimos atingir foi 59%. Abaixo, vão algumas dicas que descobrimos.

– Você deve atender aos chamados para as missas e refeições. Caso contrário, será expulso.

Se você for pego andando pela Abadia à noite, também será expulso. Não pegue o livro sem as luvas, ou morrerá.

– Com a chave existente na porta da biblioteca, poderá abrir a porta "A".

– Há uma passagem secreta na cozinha, por dentro do forno (indicada por uma seta).

– A biblioteca è totalmente escura. Só pudemos entrar lá no sexto dia, mas deve haver um modo de entrar antes.

– Do 2º para o 3º dia, aparece na mesa em frente ao forno, na cozinha, uma lamparina. Para pegá-la, você deve fazer o garoto que anda com você passar perto dela.

– A porta "E" é fechada logo após a chamada para a 2ª mesa.

Ordem dos acontecimentos

DIA 1

O padre vai recebê-lo na entrada e colocá-lo-á a par dos acontecimentos.
Missa.

DIA 2

Missa. O padre dirá que Venâncio morreu.
Após a missa, o padre chama-lo-á e dirá que só Malaquias pode entrar na biblioteca.
Severino procura-lo-á e dirá que acontecem coisas estranhas na Abadia.
Refeição.
Missa.

DIA 3

Missa. O padre dirá que Benegário desapareceu.
Após a missa, o padre leva-los-á ao homem mais velho e sábio da Abadia.
Refeição.
Missa.

DIA 4

Missa. O padre dirá que Benegário foi achado morto.
Após a missa, o padre dir-lhe-á para parar as investigações, pois Bernado Guy chegou.
Um dos freis procura-lo-á e dirá que Benegário tinha manchas negras na língua e nos dedos.
Refeição.
Missa.

DIA 5

– Um frei lhe diz que achou um estranho livro em seu quarto
– Missa
– O padre lhe dirá que Bernardo abandonará a Abadia.
– Refeição
– O padre o manda segui-lo, juntos, encontram Severino morto.
– Missa, Malaquias morre.

DIA 6

– Missa
– O padre lhe diz que deverá abandonar a Abadia no dia seguinte.
– Refeitório
– Missa

DIA 7

– Missa
– O padre lhe diz que deve ir embora. ■

SÍMBOLOS NO MAPA

- CORRIMÃO
- PAREDE
- ---- PASSAGENS
- → PASSAGEM DE ANDAR
- LIVRO
- MANUSCRITO
- CHAVE
- ARCOS
- LUVAS
- R REFEITÓRIO
- C COZINHA
- IGREJA
- ALTAR
- Q QUARTO
- PILASTRAS
- PORTA ABERTA
- PORTA TRANCADA

2º

1 →

2 →

3 ↑

E

B

OBS.: AS ESCADAS NÃO ESTÃO REPRESENTADAS NO MAPA.

POSIÇÃO À MESA

VOCÊ

GAROTO

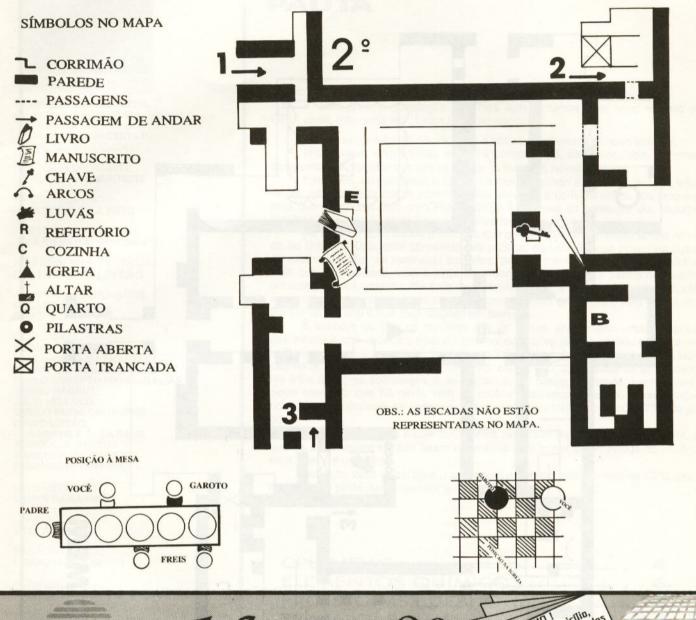
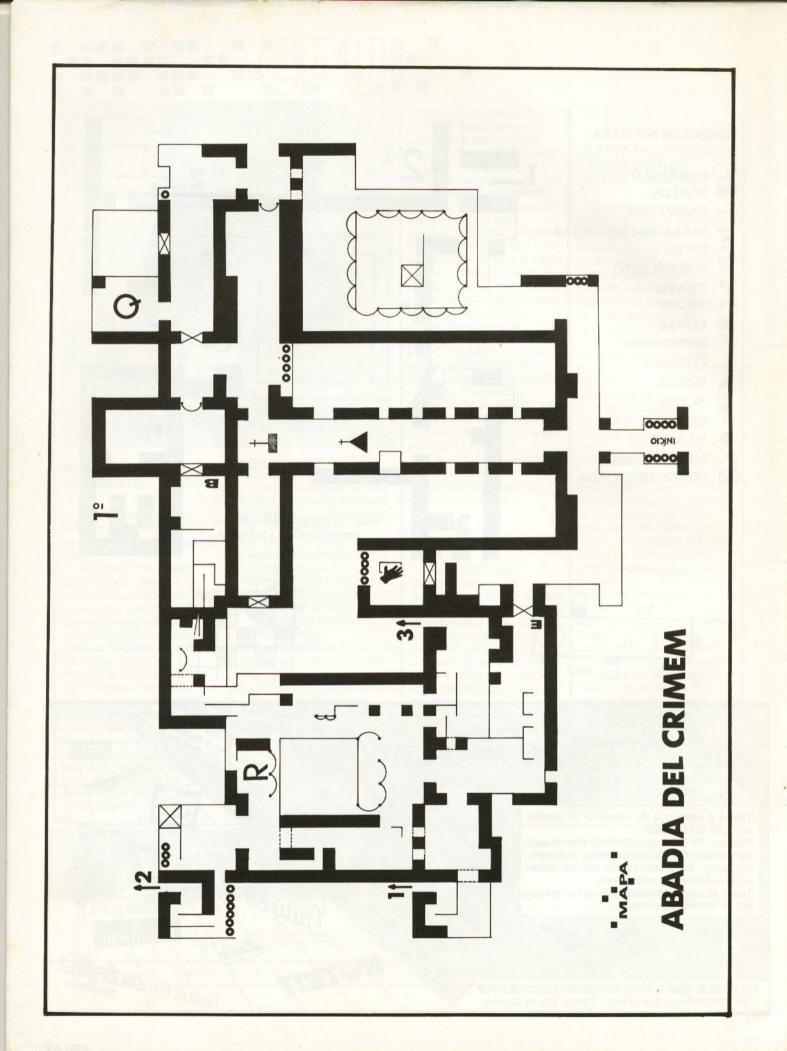
PADRE

FREIS

GAROTO

VOCÊ

POSIÇÃO NA IGREJA

1º
Q
B
R
↑2
1↑
3↑
E
INÍCIO
MAPA
ABADIA DEL CRIMEM

## AGORA TAMBÉM EM KIT (LIVRO + DISQUETE)

**LANÇAMENTO**

+
\+ 50 DICAS PARA MSX

+
100 DICAS PARA MSX

+
ASTROLOGIA NO MSX

+
CIRCUITOS ELETRÔNICOS

## LIVROS "SOFTWARE" PARA O SEU MSX !

CURSO DE MÚSICA PARA MSX

CURSO DE BASIC MSX VOL.1

DESENHOS BÁSICOS PARA MSX

COLEÇÃO DE PROGRAMAS VOL.1

COLEÇÃO DE PROGRAMAS VOL.2

LINGUAGEM DE MÁQUINA MSX

HOTLOGO

PROG. PROF. EM BASIC

PROG. AVANÇADA EM MSX

COMO USAR SEU HOTBIT

USANDO O DISK DRIVE NO MSX

APROFUNDANDO-SE NO MSX

## E MAIS...

**LINGUAGEM BASIC MSX**
**DOMINANDO O EXPERT**
**HOTDATA**
**HOTPLAN**
**HOTWORD**
**JOGOS DE HABILIDADE MSX**
**SISTEMA DE DISCO PARA MSX**
**DRIVES LEOPARD DE 3 1/2"**

Nossos livros podem ser encontrados em livrarias e lojas de computação. Se o seu livreiro ou fornecedor habitual não os tiver disponíveis, entre em contato conosco pelo telefone: (011) 843-3202.

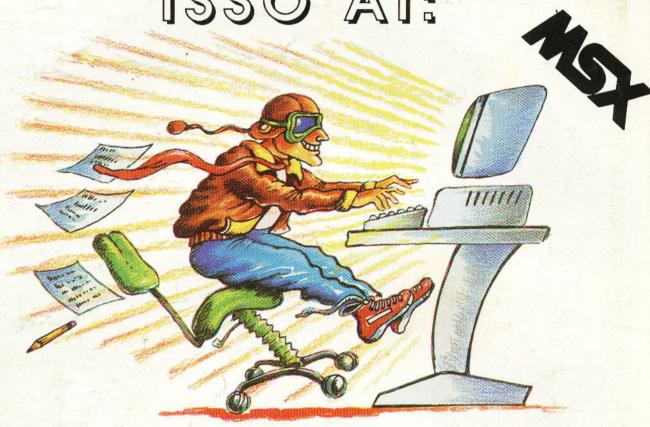
Se você não está recebendo o seu boletim gratuitamente pelo Correio, ou tem algum amigo que gostaria de recebê-lo, não deixe de enviar o cupom abaixo à **Editora Aleph, Cx. Postal 20707 CEP 01498 SãoPaulo - SP.**

NOME: ____________________

ENDEREÇO: ____________________

CEP: __________ CIDADE: __________ UF: ______

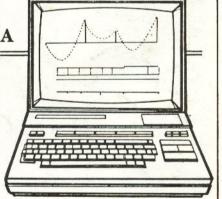
TEL:(— ) __________ MICRO: __________

# A NOVA NEWSOFT É ISSO AÍ:

A PARTIR DE AGORA O SEU MICRO FICARÁ MUITO MAIS "INTELIGENTE".

**A MELHOR E MAIS BEM APARELHADA SOFTHOUSE DO RIO DE JANEIRO, AGORA EM NOVAS INSTALAÇÕES, COLOCA A SUA DISPOSIÇÃO A MAIS COMPLETA LINHA DE PRODUTOS PARA O SEU MSX!**

**ATENÇÃO ENGENHEIROS E ESTUDANTES DE ENGENHARIA**

"Viga" é um programa para Cálculo de Estruturas Hiperestáticas (Vigas Contínuas), desenvolvido de acordo com a metodologia de ensino das escolas de Engenharia e dentro da teoria do "Método das Deformações"
Bastante simples de operar, é de extrema utilidade para Engenheiros e estudantes de Engenharia possibilitando a resolução de vigas com até 12 vãos.

Carlos Alberto Alencar Mota
Autor do programa e Engenheiro Civil (CNPq)
(Programa registrado na S.E.I.)

Breve Cálculo de Lajes, Pilares e Fundações

MSX

**Pedidos diretamente a NewSoft Informática Ltda.**
Av. Nilo Peçanha, 50 sala 906 — CEP 20.020 Rio de Janeiro - RJ - ou através de vale postal "AG. ARCOS" — cod. 522317